



Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologia
Departamento de Informática

Governança e Modelação de Processos de TI

Raquel Helena de Bragança Vasconcelos da Porciúncula
(Licenciada)

Dissertação apresentada para a obtenção do
Grau de Mestre em Engenharia Informática

Orientador: Professor Doutor Fernando Manuel Pereira da Costa Brito e Abreu

Lisboa
Fevereiro 2010

[Esta página foi intencionalmente deixada em branco]

Resumo

Estudos efectuados, ao longo dos últimos anos, têm demonstrado como os recursos de Tecnologia de Informação (TI) devem ser usados de maneira a propiciar a criação de ambientes organizacionais mais alinhados com as necessidades de negócio.

Actualmente, vivemos um processo competitivo onde as organizações gerem muitos activos – pessoas, dinheiro, instalações, fornecedores, clientes – e investem fortemente em Tecnologia de Informação. Para que a organização consiga extrair benefícios concretos do processo de manipulação do conhecimento é necessário que esta possua mecanismos estruturais de gestão que permitam criar uma base de conhecimento adequada, levando a organização a preservar o seu capital intelectual.

A governança de TI, na forma como é debatida actualmente no mercado, está mais focada no uso de boas práticas representadas por metodologias como CobiT (*Control Objectives for Information and related Technology*) e ITIL (*Information Technology Infrastructure Library*). No entanto, neste estudo vamos mais além e apresentamos uma visão de como a governança de TI pode ser útil ao alinhamento do negócio com a TI, através do uso de uma nova técnica que permite identificar os níveis de aptidão da gestão de serviços de TI, de modo a avaliar e gerir, não só os problemas complexos dos sistemas de informação e a tecnologia das organizações, mas também questões de gestão, operacionais e culturais.

Esta dissertação propõe um método para avaliar o grau de maturidade da governança de TI dentro de uma organização. Simultaneamente, foram recolhidas métricas da complexidade dos processos de TI, as quais foram correlacionadas com os níveis de maturidade. Baseado nestas correlações, foi criado um modelo que interliga a governança de TI com o nível de complexidade dos processos de TI.

Começamos por fornecer uma panorâmica sobre o trabalho que tem sido desenvolvido nas áreas de domínio desta dissertação; apresentamos alguns conceitos básicos sobre a governança de TI e qual a sua importância; relatamos detalhadamente a abordagem adaptada para determinar o perfil de governança de TI das empresas portuguesas; formalizamos um conjunto de indicadores quantitativos que permitem exprimir a complexidade de um processo, com recurso à linguagem OCL (*Object Constraint Language*), sobre um metamodelo BPMN (*Business Process Modeling Notation*); analisamos os resultados e apresentamos as conclusões. Esta dissertação termina com a identificação das linhas de orientação gerais para a continuação futura do trabalho desenvolvido.

Palavras-chave:

Governança de TI; perfil de governança de TI; boas práticas; BPMN; métricas de complexidade; OCL; complexidade de processos de TI.

[Esta página foi intencionalmente deixada em branco]

Abstract

Studies over the past years have shown how Information Technology (IT) resources should be used in order to facilitate the creation of organizational environments more closely aligned with business needs.

Today, we live a competitive process where organizations manage many assets - people, money, facilities, suppliers, customers - and invest heavily in information technology. So that organization can extract any concrete benefits of the knowledge manipulation process is necessary that there be management structural mechanisms, to enable an adequate knowledge base, leading the organization to retain its intellectual capital.

IT governance, in the way it is currently debated in the market, is more focused on the use of best practices represented by frameworks such as CobiT (Control Objectives for Information and related Technology) and ITIL (Information Technology Infrastructure Library). However, in this study, we go further and present a vision of how IT governance can be useful to align business with IT through a new technique to identify the aptitude levels of IT service management, in order to evaluate and manage, not only the complex problems of information systems and the organizations technology, but also management, operational and cultural issues.

This thesis proposes a method to evaluate the IT governance maturity degree within an organization. Simultaneously, we collected metrics of the IT processes complexity, which were correlated with the maturity levels. Based on these correlations, a model was created that connects IT governance with the IT processes complexity level.

We begin by give an overview of the work that has been developed in the areas surrounding to this thesis; we present some basic concepts of IT governance and what is its importance; we report in detail the approach adapted to determine the IT governance profile of portuguese companies; we formalized a set of quantitative indicators that allow to express the process complexity, using the OCL (Object Constraint Language) language, on a BPMN (Business Process Modeling Notation) metamodel; we analyzed the results and present the conclusions. This dissertation ends with the identification of general guidelines for the future continuation of the work developed.

Keywords:

IT governance; IT governance profile; best practices; BPMN; complexity metrics; OCL; IT process complexity.

[Esta página foi intencionalmente deixada em branco]

Agradecimentos

Uma dissertação, apesar do processo solitário a que estamos destinados, reúne contributos de várias pessoas. Desde o início do mestrado, contei com a confiança e o apoio de inúmeras pessoas e instituições. Sem estes contributos, esta investigação não teria sido possível.

Ao Professor Doutor Fernando Brito e Abreu, meu orientador, pela competência com que orientou esta minha dissertação e o tempo que generosamente me dedicou transmitindo-me os melhores e mais úteis ensinamentos, com paciência, clareza e confiança. Obrigada pela sua crítica sempre tão atempada, como construtiva. Acima de tudo, obrigada por acreditar em mim, me mostrar o caminho da Ciência e por estimular o meu interesse pelo conhecimento e pela vida académica.

Ao Dr. Rui Soares, *Business Developer* na GFI Portugal, pela sua ajuda e apoio, pela sua disponibilidade para me ensinar, pela partilha do saber e pelas valiosas contribuições para o trabalho. Muito obrigada pela sua participação, por contribuir para o meu crescimento profissional e por ser também um exemplo a ser seguido. Bem-haja.

Um sentido agradecimento ao Dr. João Lourenço, *Financial Services Sector Manager* na GFI Portugal, pelos valiosos conselhos, pela boa vontade e pelo incentivo amigo com que sempre pude contar; o meu muito obrigada pela amizade e apoio que me deu.

A todas as pessoas da GFI Portugal, que de uma forma directa ou indirecta tornaram possível o meu trabalho, em especial, aos meus colegas de trabalho António Estima, Nuno Alexandre, Nuno Lopes e Paulo Santiago que participaram directamente deste trabalho e me ajudaram em todos os momentos. A todos eles um agradecimento muito especial.

À Professora Olga Romão Branco do Instituto Superior de Psicologia Aplicada (ISPA), pelo seu precioso conhecimento e esclarecimentos sobre a construção de inquéritos que contribuíram para a realização desta dissertação.

Durante a elaboração desta dissertação, pude participar num grupo de investigadores da área de Engenharia de Software, o que representou uma oportunidade ímpar de crescimento académico e também pessoal. A todo o grupo QUASAR (*QUAntitative Approaches on Software Engineering And Reengineering*), em particular ao Anacleto Correia, João Caldeira, Jorge Freitas, José Costa, Luís Silva e Sérgio Bryton obrigada pela oportunidade de aprender e contribuir. Muito obrigada pelas oportunas manifestações de companheirismo e de encorajamento. Um sentido agradecimento ao Jorge Freitas e ao José Costa, pelo contributo fundamental que prestaram a este estudo.

Ao *itSMF (The IT Service Management Forum) Portugal* e ao Dr. Luís Vidigal, Presidente do *itSMF Portugal*, pelo inestimável contributo prestado, nesta investigação, tendo sido fundamental na nossa recolha de informação através da distribuição e preenchimento de um questionário.

Este trabalho não poderia ter sido realizado sem o apoio de toda a minha família pelo seu suporte, educação e constante presença na minha formação enquanto ser humano. À minha Avó, Dra. Maria da Glória Carvalho, pela qual nutro um profundo sentimento de amizade e gratidão e que me motiva a continuar sempre em frente, a contornar as dificuldades e a efectuar melhores escolhas; pelo inestimável apoio que me dá, pela sua bondade e por tudo o que representa para mim. Ao meu namorado, Pedro Lindo, que com o seu amor e amizade, me deu sempre coragem e confiança para continuar e que têm sido o meu porto seguro em todas as minhas aventuras.

O meu profundo e sentido agradecimento a todas as pessoas que contribuíram para a concretização desta dissertação, estimulando-me intelectual e emocionalmente.

Índice de Matérias

1 INTRODUÇÃO	2
1.1 CONTEXTO E MOTIVAÇÃO.....	2
1.2 IDENTIFICAÇÃO DO PROBLEMA.....	5
1.3 SOLUÇÃO APRESENTADA	6
1.4 PRINCIPAIS CONTRIBUIÇÕES ESPERADAS	8
1.5 ORGANIZAÇÃO DA DISSERTAÇÃO	9
2 ESTADO DA ARTE.....	12
2.1 INTRODUÇÃO	12
2.2 DESCRIÇÃO DA TAXIONOMIA PROPOSTA	12
2.3 ENQUADRAMENTO DO TRABALHO RELACIONADO	15
2.3.1 <i>Governança de TI</i>	15
2.3.1.1 Avaliação do Estado da Arte.....	15
2.3.1.2 Revisão dos Artigos de Governança de TI	15
2.3.2 <i>Maturidade de Processos de TI</i>	23
2.3.2.1 <i>Processos de Gestão de Serviços de TI</i>	24
2.3.2.1.1 Avaliação do Estado da Arte.....	24
2.3.2.1.2 Revisão dos artigos Processos de Gestão de Serviços de TI	25
2.3.2.2 <i>Processos de Desenvolvimento</i>	30
2.3.2.2.1 Avaliação do Estado da Arte.....	30
2.3.2.2.2 Revisão dos Artigos Processos de Desenvolvimento.....	31
2.4 RESUMO DA TAXIONOMIA E DO ESTADO DA ARTE.....	36
3 GOVERNANÇA DE TI	40
3.1 INTRODUÇÃO	40
3.2 A IMPORTÂNCIA DA GOVERNANÇA DE TI	41
3.3 COMO AS EMPRESAS GOVERNAM A TI: DECISÕES-CHAVE E ARQUÉTIPOS DE TI	43
3.4 COMO É QUE AS DECISÕES SÃO FORMADAS E DIVULGADAS.....	47
3.5 PERFIS DE GOVERNANÇA DE TI.....	49
3.6 ABORDAGENS DA GOVERNANÇA DE TI	50
3.7 PLANEAR E AVALIAR A GOVERNANÇA DE TI	54
3.8 RECOMENDAÇÕES PARA A GOVERNANÇA DE TI.....	55
4.AVALIAR A GOVERNANÇA DE TI.....	58
4.1 QUESTIONÁRIO SOBRE A GOVERNANÇA DE TI	58
4.2 COMO AS EMPRESAS GOVERNAM A TI: DECISÕES-CHAVE E ARQUÉTIPOS DE TI	59
4.3 FACTORES DE VARIAÇÃO NOS PERFIS DA GOVERNANÇA DE TI	63
4.4 PERFIS DA GOVERNANÇA DE TI	64
4.5 ABORDAGENS DA GOVERNANÇA DE TI	66
5 COMPLEXIDADE DE UM PROCESSO	72
5.1 INTRODUÇÃO	72
5.1.1 ABORDAGEM METODOLÓGICA	72
5.2 METAMODELO	74
5.3 OCL	76
5.4 RESTRIÇÕES OCL	77
5.4.1 MÉTRICAS COMPLEXAS	77
5.4.1.1 TAMANHO DO PROCESSO/MODELO	78
5.4.1.2 COMPLEXIDADE DO PROCESSO/MODELO	79
5.4.1.3 ESTRUTURA DO PROCESSO/MODELO.....	84
5.4.1.4 MODULARIZAÇÃO DO PROCESSO/MODELO	86
5.4.1.5 MÉTRICAS NOVAS PROPOSTAS.....	88

5.5 USE	94
6 ANÁLISE DOS RESULTADOS	102
6.1 INTRODUÇÃO	102
6.2 ANÁLISE DE CORRELAÇÃO	102
6.2.1 INTRODUÇÃO	102
6.2.2 ANÁLISE DE RESULTADOS.....	103
6.3 CASO DE ESTUDO	105
6.3.1 CARACTERIZAÇÃO DO ORGANISMO.....	105
6.3.2 ANÁLISE E RESULTADOS DO QUESTIONÁRIO DE GOVERNANÇA DE TI	106
6.3.3 COMPLEXIDADE DOS PROCESSOS.....	111
6.3.4 CONCLUSÕES DO CASO DE ESTUDO	114
6.4 IMPACTO DA GOVERNANÇA NOS PROCESSOS	115
7 CONCLUSÕES E TRABALHO FUTURO.....	120
7.1 CONCLUSÕES.....	120
7.2 EVOLUÇÃO FUTURA	125
A. INQUÉRITO DE GOVERNANÇA DE TI.....	128
A) INQUÉRITO SOBRE GOVERNANÇA DE TI	128
B) RESULTADOS DO INQUÉRITO SOBRE GOVERNANÇA DE TI	138
C) INTERPRETAÇÃO DOS RESULTADOS DO INQUÉRITO SOBRE GOVERNANÇA DE TI.....	158
B. METAMODELO BPMN E RESTRIÇÕES OCL.....	168
A) METAMODELO BPMN.....	168
B) MÉTRICAS OCL	177
BIBLIOGRAFIA	203

Índice de Figuras

FIGURA 1.1- LINHAS VERTICAIS (GESTÃO) E HORIZONTAIS (PROCESSOS) DE UMA ORGANIZAÇÃO (EXTRAÍDO DE [BON, 2008]).....	3
FIGURA 1.2- GOVERNANÇA DE TI VERSUS GESTÃO DE TI, SEGUNDO SOHAL & FITZPATRICK [SOHAL AND FITZPATRICK, 2002] EXTRAÍDO DE [BON, 2008]	4
FIGURA 1.3- ROADMAP GRÁFICO	8
FIGURA 2.4- A POSIÇÃO DO DOMÍNIO DE GESTÃO DE INFORMAÇÃO ENTRE O NEGÓCIO E A TECNOLOGIA DE INFORMAÇÃO NO MODELO SAME (EXTRAÍDO DE [BON, 2008])	17
FIGURA 2.5- INTEGRAÇÃO DOS DOIS MODELOS (EXTRAÍDO DE [PAULK ET AL., 1995])	35
FIGURA 3.6- OS TRÊS PERFIS MAIS EFICAZES MEDIDAS PELO DESEMPENHO DA GOVERNANÇA DE TI (ADAPTADO DE [WEILL, 2004]) ..	49
FIGURA 4.7- ALGUNS DADOS SOBRE A AMOSTRA DE EMPRESAS	60
FIGURA 4.8- DESEMPENHO DA GOVERNANÇA DE TI PARA A MÉDIA DOS 52 INQUÉRITOS, PARA AS PEQUENAS, MÉDIAS E GRANDES EMPRESAS CONSTANTES DA AMOSTRA.....	67
FIGURA 5.9- FASE DE METAMODELAÇÃO, TRANSFORMAÇÃO E INSTANCIAÇÃO	74
FIGURA 5.10- CLASSES DO METAMODELO BPMN.....	75
FIGURA 5.11- TIPOS DEFINIDOS NA BIBLIOTECA PADRÃO OCL	76
FIGURA 5.12 – MÉTRICA <code>CFC_XOR_DATABASED()</code> DE UM PROCESSO.....	80
FIGURA 5.13 – MÉTRICA <code>LENGTH()</code> DE UM PROCESSO	82
FIGURA 5.14– MÉTRICA <code>CNC</code> DE UM PROCESSO.....	84
FIGURA 5.15– MÉTRICA <code>NESTING_DEPTH</code> DE UM PROCESSO.....	85
FIGURA 5.16– MÉTRICA <code>HKM</code> DE UM PROCESSO	87
FIGURA 5.17 - FUNÇÃO QUE CALCULA TODOS OS CAMINHOS POSSÍVEIS DANDO UMA ORIGEM E UM DESTINO.....	88
FIGURA 5.18 – FUNÇÕES QUE NOS PERMITEM SABER INFORMAÇÃO SOBRE O MAIOR E MENOR CAMINHO DE UM PROCESSO	89
FIGURA 5.19 – FUNÇÕES QUE NOS PERMITEM SABER INFORMAÇÃO SOBRE O NÚMERO DE CAMINHOS POSSÍVEIS E OS <i>FLOWOBJECTS</i> QUE ESTÃO CONTIDOS NESSES CAMINHOS	90
FIGURA 5.20 – FUNÇÕES QUE NOS PERMITEM SABER INFORMAÇÃO SOBRE AS ACTIVIDADES (<i>TASKS</i> E <i>SUB-PROCESSOS</i>).....	91
FIGURA 5.21 - FUNÇÕES QUE NOS PERMITEM SABER INFORMAÇÃO SOBRE AS <i>GATEWAYS</i> E <i>GATES</i> (<i>INPUT</i> E <i>OUTPUT</i>).....	92
FIGURA 5.22 - FUNÇÕES QUE NOS PERMITEM SABER INFORMAÇÃO SOBRE OS EVENTOS (<i>START</i> , <i>INTERMEDIATE</i> E <i>END EVENTS</i>).....	93
FIGURA 5.23 – ABORDAGEM USE (HTTP://WWW.DB.INFORMATIK.UNI-BREMEN.DE/PROJECTS/USE/).....	94
FIGURA 5.24 – EXEMPLO TEXTUAL USE	95
FIGURA 5.25 – JANELA “ <i>OPEN SPECIFICATION</i> ” DO USE	95
FIGURA 5.26 – “ <i>BPMN_MODEL.USE</i> ” CARREGADO NO USE	96
FIGURA 5.27 – META-OBJECTOS CARREGADOS.....	97
FIGURA 5.28 - META- LIGAÇÕES CARREGADAS	98
FIGURA 5.29 – EXEMPLIFICAÇÃO DE ALGUMAS MÉTRICAS NO USE	99
FIGURA 5.30 - EXEMPLIFICAÇÃO DE ALGUMAS MÉTRICAS COMPLEXAS NO USE	100
FIGURA 6.31 - PROGRAMA SPSS <i>STATISTIC</i>	104

Índice de Quadros

TABELA 2.1: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA.....	17
TABELA 2.2: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA.....	18
TABELA 2.3: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	19
TABELA 2.4: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	20
TABELA 2.5: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	21
TABELA 2.6: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	22
TABELA 2.7: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	23
TABELA 2.8: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	26
TABELA 2.9: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	27
TABELA 2.10: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	29
TABELA 2.11: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	30
TABELA 2.12: MÉTRICAS COMPLEXAS PARA SOFTWARE E BPMs	32
TABELA 2.13: COMPARAÇÃO ENTRE AS ABORDAGENS GAM, BSC E GQM.....	32
TABELA 2.14: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	33
TABELA 2.15: CLASSIFICAÇÃO BASEADA NOS CRITÉRIOS DEFINIDOS NA TAXIONOMIA	35
TABELA 2.16: RESUMO DO ESTADO DA ARTE	37
TABELA 3.17: 5 DOMÍNIOS DE DECISÃO DE TI (ADAPTADO DE [WEILL, 2004]).....	43
TABELA 3.18: QUESTÕES CHAVE PARA CADA DECISÃO DE TI (ADAPTADO DE [WEILL AND ROSS, 2004])	44
TABELA 3.19: ARQUÉTIPOS DE GOVERNANÇA DE TI (ADAPTADO DE [WEILL, 2003]).....	45
TABELA 3.20: MECANISMOS DE GOVERNANÇA DE TI (ADAPTADO DE [BROADBENT AND WEILL, 1998] [WEILL AND WOODHAM, 2002]).....	48
TABELA 3.21: LIÇÕES DE GOVERNANÇA PARA GESTORES DE TOPO (ADAPTADO DE [WEILL AND ROSS, 2004])	51
TABELA 3.22: <i>TRADE-OFFS</i> E O MELHOR DOS DOIS MODELOS (ADAPTADO DE [BROWN AND MAGILL, 1998] E [ROCKART, 1996]) ..	53
TABELA 3.23: AVALIAÇÃO DA EFICIÊNCIA DA GOVERNANÇA DE TI (ADAPTADO DE [WEILL AND WOODHAM, 2002] E [BROADBENT AND WEILL, 1998])	55
TABELA 4.24: ARQUÉTIPOS DE GOVERNANÇA USADOS PARA DIFERENTES TIPOS DE DECISÕES.....	61
TABELA 4.25: PERFIL DE GOVERNANÇA DE TI DAS EMPRESAS PORTUGUESAS.....	64
TABELA 4.26: COMPARAÇÃO DO PERFIL DE GOVERNANÇA DE TI DAS EMPRESAS CONSTANTES DA AMOSTRA COM OS TRÊS MELHORES TIPOS DE DESEMPENHO IDENTIFICADOS EM [WEILL, 2004]	65
TABELA 4.27: PERFIL DE GOVERNANÇA DE TI DAS PEQUENAS, MÉDIAS E GRANDES EMPRESAS CONSTANTES DA AMOSTRA	65
TABELA 4.28: COMPARAÇÃO DO PERFIL DE GOVERNANÇA DE TI DAS PEQUENAS, MÉDIAS E GRANDES EMPRESAS CONSTANTES DA AMOSTRA COM OS TRÊS MELHORES TIPOS DE DESEMPENHO IDENTIFICADOS EM [WEILL, 2004]	66
TABELA 5.29: TIPOS COLECCÃO <i>OCL</i>	76
TABELA 5.30: MÉTRICAS COMPLEXAS	78
TABELA 6.31: CRITÉRIOS PARA ANÁLISE DOS ÍNDICES DE CORRELAÇÃO	103
TABELA 6.32: MATRIZ 8x8 COM RESULTADOS OBTIDOS ATRAVÉS DO PROGRAMA SPSS STATISTIC	105
TABELA 6.33: CARACTERIZAÇÃO DO ORGANISMO PÚBLICO EM ESTUDO.....	106
TABELA 6.34: PERFIL DE GOVERNANÇA DE TI DO ORGANISMO EM ESTUDO	107
TABELA 6.35: IMPORTÂNCIA DOS RESULTADOS DE GOVERNANÇA DE TI	109
TABELA 6.36: INFLUÊNCIA DA GOVERNANÇA DE TI	109
TABELA 6.37: AVALIAÇÃO DOS ÓRGÃOS COM INTERVENÇÃO NA GOVERNANÇA DE TI.....	110
TABELA 6.38: MECANISMOS DE GOVERNANÇA DE TI	111
TABELA 6.39: MÉTRICAS DA GESTÃO DE INCIDENTES E GESTÃO DE PROBLEMAS.....	112
TABELA 6.40: MÉTRICAS DA GESTÃO DE INCIDENTES E GESTÃO DE PROBLEMAS.....	113
TABELA 6.41: QUADRO SÍNTESE DAS CARACTERÍSTICAS DO CASO DE ESTUDO.....	115
TABELA B.42: MÉTRICAS PARA CALCULAR O TAMANHO DE UM PROCESSO	177
TABELA B.43: MÉTRICAS PARA CALCULAR O TAMANHO DE UM MODELO	185

Glossário

ACM - *Association for Computing Machinery*
ANOVA – *Analysis of Variance*
API - *Application Programming Interface*
BPG - *Baseline Practices Guide*
BPM – *Business Process Model*
BPMN - *Business Process Modeling Notation*
BPR - *Business Process Redesign*
BSC - *Balanced Scorecard*
CFC – *Control Flow Complexity*
CIO - *Chief Information Officer*
CISR – *Center for Information System Research*
CMM - *Capability Maturity Model*
CMMI - *Capability Maturity Model Integration*
COBIT - *Control Objectives for Information and related Technology*
CORBA - *Common Object Request Broker Architecture*
COSO - *Committee of Sponsoring Organizations of the Treadway Commission*
CSCW - *Computer-Supported Cooperative Work*
CSFs – *Critical Success Factors*
DoD - *Domain of Discourse*
DOM - *Document Object Model*
IBM - *International Business Machines Corporation*
IDEF - *Integrated DEFinition methods*
IT - *Information Technology*
itSMF – *IT Service Management Forum*
IS – *Information Systems*
ISMS – *Information Security Management System*
ITG - *Information Technology Governance*
ITIL - *Information Technology Infrastructure Library*
ITOMAT - *Information Technology Organization Modeling and Assessment Tool*
ITSM – *IT Service Management*
GAM – *Goal, Attribute, Measure*
GQM - *Goal, Question, Metric*
KGI - *Key Goal Indicators*
KPI – *Key Performance Indicators*

M2DM - *Meta-Model Driven Measurement*
MIT – *Massachusetts Institute of Technology*
MOF - *Meta-Object Facility*
MOODLE - *Modular Object-Oriented Dynamic Learning Environment*
OCL - *Object Constraint Language*
OLA - *Operational Level Agreement*
OMG - *Object Management Group*
OO – *Object Oriented*
OWL - *Web Ontology Language*
QoS – *Quality of Service*
QUASAR - *QUantitative Approaches on Software Engineering And Reengineering*
ROA - *Return on Assets*
ROE – *Return on Equity*
ROI - *Return on Investment*
SAME - *Strategic Alignment Model Enhanced*
SAX - *Simple API for XML*
SEI - *Software Engineering Institute*
SI – *Sistema de Informação*
SLA – *Service Level Agreement*
SLM – *Service Level Management*
SOX – *Sarbanes-Oxley act*
SPI - *Software Process Improvement*
SPICE - *Software Process Improvement and Capability dEtermination*
TI – *Tecnologia de Informação*
TQM - *Total Quality Management*
UML - *Unified Modeling Language*
USE - *UML-Based Specification Environment*
XMI - *XML Metadata Interchange*
WfMS - *Workflow Management System*

Capítulo 1

Introdução

Conteúdo

1 INTRODUÇÃO	2
1.1 CONTEXTO E MOTIVAÇÃO.....	2
1.2 IDENTIFICAÇÃO DO PROBLEMA.....	5
1.3 SOLUÇÃO APRESENTADA	6
1.4 PRINCIPAIS CONTRIBUIÇÕES ESPERADAS.....	8
1.5 ORGANIZAÇÃO DA DISSERTAÇÃO	9

Este capítulo introduz os conceitos principais desta dissertação, descreve uma motivação para a existência de uma relação entre o modelo de governança adaptado por uma organização e a complexidade dos seus processos e enumera os principais objectivos e contribuições. No final, faz uma apresentação sumária dos conteúdos presentes nos restantes capítulos do documento.

1 Introdução

1.1 Contexto e Motivação

As organizações não têm todas os mesmos objectivos. Dependendo da sua natureza e dos seus objectivos, o modelo de gestão precisa de ser configurado de tal forma que resulte num desempenho máximo. As diferenças da sua natureza podem ser diversas, desde variações da dimensão de uma organização, até à sua cultura, tipo de produto ou serviço. Os objectivos das organizações também podem diferir da mesma forma, variando de lucrativas para não-lucrativas, voltadas para a penetração do mercado e margens de lucro máximas, ou, simplesmente, para criar emprego. Dependendo do tipo de organização, a escolha do modelo de gestão será feita com base nos processos para a realização dos objectivos da organização. O peso da gestão de processos na gestão do sistema de informação pode diferir consideravelmente de organização para organização [Boes, 2007]. A organização de uma empresa de serviço de TI deve adequar-se aos seus objectivos. Existem vários estudos organizacionais que tendem a concentrar-se sobre a chamada linha hierárquica organizacional, ilustrada na figura 1.1 [Bon, 2008]. Nesta aproximação, as linhas verticais são muitas vezes o principal ponto de partida. A questão é saber se estas linhas verticais nos permitem perceber os objectivos da nossa organização. A organização das linhas verticais é caracterizada pela estrutura dos departamentos, sendo estes constituídos por pessoas com capacidades e qualidades suficientes para executarem actividades específicas. Geralmente, quando uma organização tem vários departamentos, nenhum departamento é capaz de realizar todos os objectivos da organização por si próprio [Bon, 2008]. Tipicamente, os departamentos de TI têm tarefas específicas, como por exemplo, executar tarefas de gestão de uma aplicação ou de *helpdesk*¹. Assim, para a realização dos objectivos da organização, é necessário que estas tarefas específicas estejam alinhadas. A regularização e organização destas tarefas são feitas através da introdução de processos, que são representados pelas linhas horizontais.

Um processo é definido como: "Um conjunto estruturado de actividades concebidas para realizar um objectivo específico" [ITIL v3, 2007]. O foco está, então, nos processos organizados horizontalmente (como ilustrado na figura 1.1), que pretendem entregar algo para um cliente (interno ou externo) de uma maneira eficiente e eficaz. A organização de um processo pode aparecer tanto numa forma formalizada como numa forma não formalizada. A organização do processo não formalizado é baseada num conjunto de acordos informais (colaboração) entre os departamentos e os funcionários, com ou sem requisitos de qualidade especificados. Estes acordos podem ser incorporados em diversos procedimentos.

¹ **Help Desk** – “Um ponto de contacto para os utilizadores registarem incidentes. Usualmente, um *help desk* é mais focado a nível técnico do que um *Service Desk* e não fornece um único ponto de contacto para todas as interacções. O termo *help desk* é frequentemente usado como sinónimo de *Service Desk*.”

Service Desk – “O único ponto de contacto entre o prestador de serviço e o utilizador. Um *Service Desk* típico gere incidentes e pedidos de serviço.” [Iqbal, 2007]

A organização formalizada do processo é caracterizada por uma combinação de sub-processos detalhados que, por sua vez, estão focados na realização dos objectivos da empresa.

Normalmente, as organizações são uma mistura das duas dimensões, horizontal e vertical, de gestão.

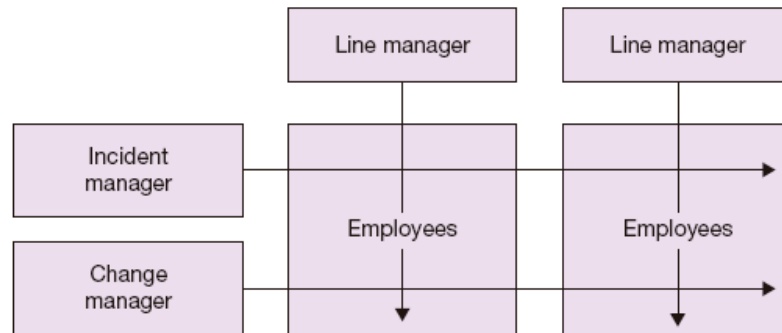


Figura 1.1: Linhas verticais (Gestão) e horizontais (Processos) de uma organização (extraído de [Bon, 2008]).

Neste ponto, interessa questionar como é que as linhas verticais devem gerir os processos, se isto é realmente necessário e, em caso afirmativo, como é que se organiza essa gestão.

Dentro de cada organização ambas as dimensões estão presentes. A dimensão de gestão descreve quem executa as actividades, ao passo que a dimensão do processo descreve a forma como são realizados os objectivos da organização.

Para o correcto funcionamento de uma organização, é necessário ter uma visão clara sobre o papel dos processos, assim como, saber como é que esses processos serão geridos. É ainda mais importante compreender e aceitar as consequências dessas escolhas. Dependendo das escolhas feitas, uma organização será completamente ou parcialmente gerida através de processos [Bon, 2008].

Assim, quando falamos em aspectos como a gestão, o controlo e a medição dos sistemas de informação, podemos falar em Governança da Tecnologia da Informação (ITG). A governança de TI, que foi um conceito relativamente novo no final dos anos 90, tem ganho importância no século XXI devido a factores como o colapso da *Enron Inc* [Enron_scandal, 2009] e a necessidade de um sistema de comunicação e divulgação financeiro melhor, como solicitado pela Presidência da Comissão de Títulos e Câmbios dos E.U. em 2001 [Pitt, 2001]. Posteriormente, legislações como a Lei *Sarbanes-Oxley* (SOX) [Sarbanes Oxley, 2002], nos Estados Unidos, e *Turnbull Guidance* [FRC, 2005], no Reino Unido, deram um novo impulso para a necessidade da ITG. Outros factores que fazem com que as empresas dêem mais importância à gestão, ao controlo e à medição dos sistemas de informação incluem o risco associado à informação, os investimentos realizados pelas empresas em recursos de TI e a necessidade de se ser competitivo no mercado. Todos esses factores enfatizam a necessidade de medir o desempenho ou a eficácia dos sistemas de informação [Nicho and Cusack, 2007].

A governança de TI fornece mecanismos que permitem à gestão de SI/TI desenvolver e planear negócios de TI, atribuir responsabilidades, e priorizar iniciativas de TI. Existe uma clara distinção entre governança e gestão. A governança é essencialmente estratégia e a gestão é essencialmente tática, sugerindo que a governança permite a criação de um modelo no qual outros conseguem gerir

eficazmente as suas funções. A governança de TI não consiste na tomada de decisões específicas sobre TI, mas sim em administrá-la e utilizá-la para a concretização dos objectivos de desempenho de âmbito corporativo da organização [Charan, Ram - 2005]. A governança de TI integra e institucionaliza as melhores práticas de planeamento, organização, aquisição, execução e monitorização do desempenho de TI, assim como a entrega e suporte aos serviços implementados. A governança de TI também permite que sejam efectuadas avaliações periódicas, de modo que as vantagens sobre a informação obtida permitam maximizar benefícios e capitalizar oportunidades e vantagens competitivas [Weill and Ross, 2004].

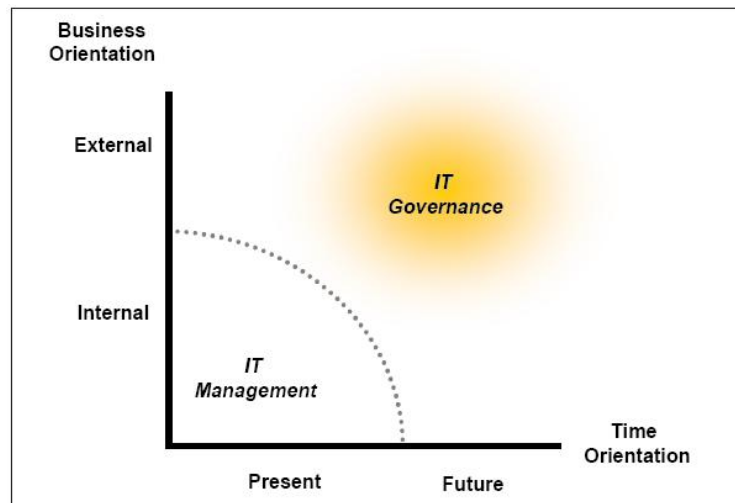


Figura 1.2: Governança de TI versus Gestão de TI, segundo Sohal & Fitzpatrick [Sohal and Fitzpatrick, 2002] extraído de [Bon, 2008]

Assim sendo, a avaliação de maturidade de governança é essencial para a boa monitorização, valorização e gestão de processos e estruturas de TI. Em particular, usando um método para avaliar a maturidade da governança de TI, é possível comparar e racionalmente escolher entre potenciais cenários futuros [Johnson, 2008]. Um exemplo seria a transferência do processo de autoridade para a tomada de decisão para aquisição de produtos de software do nível da unidade de negócio para o nível de operações de TI. Aqui, interessa questionar como é que isto iria melhorar a complexidade dos processos afectados. A possibilidade de realizar uma análise entre potenciais cenários é um dos mais importantes benefícios de ter em vigor um método de avaliação de governança eficaz de TI. Outro benefício é a possibilidade de obter um padrão de referência em relação a outras organizações. Além disso, um método de avaliação de boa governança de TI pode fornecer resultados prescritivos úteis sobre o que pode ser feito para melhorar a governança de TI dentro da organização sob avaliação [Johnson, 2008].

É interessante questionar, se existe, então, alguma relação entre a maturidade da governança e a complexidade dos processos de uma organização e se é possível perceber de que forma os processos se adequam a um determinado tipo de governança.

1.2 Identificação do Problema

Medir a eficácia dos Sistemas de Informação (SI) tem sido uma das principais preocupações dos gestores de SI e dos gestores corporativos desde a década de 80 [Nicho and Cusack, 2007]. Um estudo conduzido por Dickson e seus colegas sobre as principais preocupações dos sistemas de informação, revelou que “medir e melhorar a eficácia/produtividade dos SI” ficou classificado em quinto lugar, entre os principais problemas enfrentados por uma organização [Dickson, 1984]. Noutro estudo, realizado em meados dos anos 80, por Brancheau e Wetherbe, “medir a eficácia dos sistemas de informação” ficou classificado em nono lugar na lista das principais questões dos gestores dos sistemas de informação e em quarto lugar para gestores, no geral [Brancheau, 1987]. No virar do milénio passado, Markus e os seus colegas, salientaram que o sucesso dos SI é um dos mais duradouros temas de investigação na área de Sistemas de Informação² [Marcus, 2000]. "A medição do sucesso dos sistemas de informação tem permanecido na agenda da investigação há mais de trinta anos" [Zviran, 2003]. Um estudo mais recente realizado pela *Pricewaterhouse Coopers*, patrocinado pelo Instituto de Governança de TI (citado em [Sraeel, 2004]), sobre uma amostra de 7.000 entrevistados, constatou que um dos dez maiores problemas citados por estes inquiridos foi a "falta de visão de quão bem a TI está a ser desempenhada", e ainda, 80 por cento desta amostra são da opinião de que a governança de TI ou algum tipo de mecanismo de governança é necessário para resolver este problema [Nicho and Cusack, 2007].

Actualmente, existem alguns métodos de avaliação da maturidade da governança de TI. Por exemplo, Weill e Ross desenvolveram um referencial (*framework*) de governança de TI baseado em apenas algumas perguntas que podem ser usadas para identificar responsabilidades para a tomada de decisão de alto nível. Contudo o seu trabalho não deu mais orientação sobre a forma como a organização de TI deve realmente realizar o seu trabalho [Weill and Ross, 2004].

A norma ISO/IEC 20000 [International Organization for Standardization, 2005] e o ITIL (*Information Technology Infrastructure Library*) [Office of Government Commerce, 2003] ajudam na criação de processos relacionados com a entrega³ e o suporte⁴ de serviços. O ITIL detalha ainda a criação e a manutenção dos acordos de níveis de serviços (SLA) e dos acordos dos níveis de operação (OLA). No entanto, nem a referida norma, nem o ITIL, fornecem apoio estratégico às preocupações de TI. McLean e Soden [McLean et al., 1977], citados em [O'Connor, 1993] dividem a gestão de sistemas de

² Entendida como área científica com um corpo de conhecimento próprio, tal como reconhecido e proposto em 1998 pela ACM (*Association for Computing Machinery*) [[ACM Computing Classification System](#), 2009].

³ Exemplos de processos de entrega de serviços ITIL v2 e da norma ISO 20000 são: Gestão da Capacidade, Gestão da Disponibilidade, Gestão da Continuidade, Gestão da Segurança, etc.

⁴ Exemplos de processos de suporte de serviços ITIL v2 e da norma ISO 20000 são: Gestão de Alterações, Gestão de Configurações, Gestão de Incidentes, Gestão de Problemas, etc.

informação em três níveis⁵: operacional, tático e estratégico. A gestão de informação a um nível tático e, sobretudo, a um nível estratégico permite enquadrar a informação no contexto global da empresa e obter vantagens [O'Connor, 1993]. Uma das principais razões que justificam levar a cabo estratégias de gestão de informação é a melhoria das capacidades de decisão por parte da gestão [Marchand, 1990]. Segundo [O'Connor, 1993], empresas com planos estratégicos para os sistemas de informação bem integrados com a gestão, conseguem, sistematicamente, melhores resultados do que aquelas que não os têm.

O problema subjacente a esta dissertação é assim o do alinhamento entre os níveis estratégico, tático e operacional que a literatura indica ser um factor crítico de sucesso para obter sistemas de informação mais eficazes. Se esse alinhamento existir, então o modelo de governança de TI deverá ter impacto na forma como os processos de TI da organização estão formalizados. Como veremos, o modelo de governança de TI pode caracterizar-se através de um perfil construído com base num conjunto de arquétipos. É expectável que perfis diferentes de governança tenham consequência na forma como uma organização adopta boas práticas na definição dos seus processos de TI, usando ou não enquadramentos de referência como o ITIL ou o CobiT. Estes enquadramentos são propostos na literatura como veículos para conseguir uma maior maturidade na gestão de serviços de TI, através da definição e orquestração de um conjunto de processos. Processos mais maduros implicam mais actividades, mais pontos de controlo, mais papéis envolvidos, logo mais circuitos de circulação de informação (*workflows*) e mais decisões a serem tomadas. Como consequência, os modelos desses processos serão mais complexos. Esta dissertação pretende assim contribuir para uma avaliação desta teoria. Resumidamente, esta dissertação visa contribuir para a avaliação do impacto que o perfil de governança de TI tem na complexidade dos seus processos de TI.

1.3 Solução Apresentada

Diversos referenciais de TI têm sido desenvolvidos para fornecer orientações e melhores práticas para a indústria das TI [Sallé, 2004]. Na essência, estes referenciais abordam tanto o domínio da governança de TI (por exemplo o referencial de *Weill e Ross* [Weill and Ross, 2004]), como o domínio

⁵ Na gestão de uma unidade de negócio, que tem por base a obtenção e utilização de recursos de forma eficiente, para se atingir os objectivos organizacionais, é necessária informação a três níveis: estratégico, operacional e tático. Neste sentido, à medida que descemos na pirâmide hierárquica organizacional a especificidade aumenta, pois é necessário resolver problemas mais específicos de determinada tarefa, enquanto ao nível de topo as preocupações são mais gerais, afectando a generalidade das funções da organização:

Nível Estratégico (nível de topo) - São tomadas decisões estratégicas; são complexas e exigem informação bastante variada e ao nível das relações da organização/meio envolvente, não se exige muita especificidade. Estão incluídas nela a definição dos objectivos e a elaboração de políticas gerais da organização. A informação provém de fontes externas à organização e também dos outros níveis hierárquicos.

Nível Tático (nível intermédio) - Onde têm lugar as decisões táticas e que exigem informação pormenorizada, com alguma triagem, havendo responsabilidades na interpretação da informação, que provém de fontes internas e sendo obtida com alguma frequência.

Nível Operacional (nível de base) - Aqui são tomadas as pequenas decisões ou as decisões operacionais. Decisões para problemas bem definidos cuja resolução é, muitas vezes, baseada em dados factuais programáveis e através da aplicação de rotinas informáticas. São necessárias informações pormenorizadas e bem definidas, provenientes essencialmente do sistema interno, com vista a acções imediatas. [Oliveira, 1994]

da gestão dos processos de TI (por exemplo, o ITIL). A solução apresentada nesta dissertação permite adoptar um modelo que combine as duas noções, tanto de governança de TI, como de gestão dos processos de TI, com uma visão clara sobre as preocupações e responsabilidades de cada domínio. Assim, nesta dissertação apresentamos uma proposta de uma nova técnica para relacionar a gestão de topo com a gestão processual, de modo a avaliar e gerir, não só os problemas complexos dos sistemas de informação e a tecnologia das organizações, mas também questões de gestão e operacionais [Woodhouse, 2007]. A inovação não está no modelo *per se*, dado este já ser conhecido há bastante tempo, mas no modo como se relacionam (gestão de topo com a gestão processual) e nos resultados da sua integração.

A abordagem metodológica proposta nesta dissertação começa com a pesquisa e selecção de um modelo existente sobre governança de TI. Seguidamente, elaboramos um questionário para fazer o levantamento do grau de maturidade e do perfil de governança de uma organização. Com os resultados obtidos podemos analisar e mapear os tipos de decisões existentes numa empresa, os arquétipos para a tomada dessas decisões e como é que essas decisões são tomadas e monitorizadas, através de um conjunto de mecanismos de governança.

Após a fase de levantamento/caracterização dos componentes da governança de TI, como é que eles podem ser representados e o que se faz para se obter uma governança de TI eficaz, propomos outro passo da nossa abordagem metodológica, onde o objectivo é a determinação da complexidade dos processos. Para isso, usamos um metamodelo da linguagem BPMN (*Business Process Modeling Notation*) [Freitas, 2010] de modelação de processos, robustecido com restrições (regras de boa formação) em OCL (*Object Constraint Language*) [OCL, 2003]. Este metamodelo é instanciado por forma a podermos avaliar a complexidade dos processos.

Na verdade, existem várias metodologias para modelação de processos, ferramentas de TQM (*Total Quality Management*) [Feigenbaum, 1951], *Six Sigma* [wikipedia, 2009], *Design for Six Sigma* [wikipedia, 2009], BPR (*Business Process Redesign*) [The University of British Columbia, 2009], BPM (*Business Process Management*) [wikipedia, 2009], IDEF (*integrated DEFinition Methods*) [IDEF, 2006], etc. A linguagem BPMN para modelar processos, é usada nesta dissertação, para ajudar a representar a forma como as actividades são realizadas, estabelecendo assim uma base para futuras análises sobre a forma como a organização pode melhorar.

Para melhor visualizarmos a solução que propomos, apresentamos na figura 1.3 um *roadmap* gráfico. Este *roadmap* mostra as fases necessárias para atingir o objectivo proposto:

- (i) Determinação do perfil de governança de TI – Escolha do modelo de governança de TI; definição dos arquétipos de governança de TI; distribuição do questionário através do *itSMF*; cruzamento dos resultados obtidos através do questionário com os arquétipos de governança definidos;

- (ii) Determinação da complexidade dos processos de TI⁶ – Formalização de métricas de complexidade do processo sobre o metamodelo de BPMN proposto em [Freitas, 2010]; instanciación desse metamodelo com o exemplo proposto em [Costa, 2010]; obtenção dos valores das métricas para esse exemplo;
- (iii) Análise de correlação das métricas determinadas;
- (iv) Análise de um caso de estudo completo (perfis de governança + avaliação de processos);
- (v) Verificação da existência de uma relação entre a gestão de topo e a gestão processual.

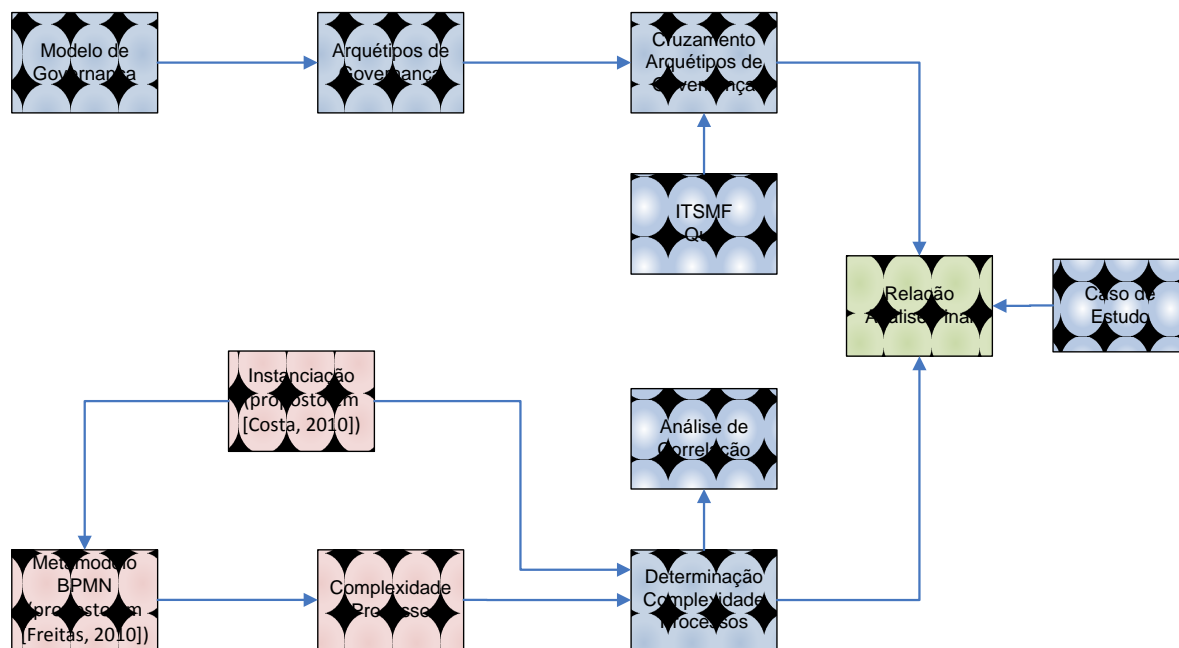


Figura 1.3: Roadmap gráfico

1.4 Principais Contribuições Esperadas

No ambiente empresarial altamente competitivo de hoje, o uso eficaz e inovador da tecnologia da informação (TI) tem o potencial de transformar o negócio, bem como afectar positivamente o desempenho das organizações. Uma série de pesquisas, mostrou que o uso intensivo da TI, que deve estar particularmente alinhado com a estratégia do negócio, pode fornecer um número de oportunidades e benefícios reais para empresas de todas as actividades e dimensões [Peppard, 2002] [Caldeira, 2001].

A principal contribuição que esperamos oferecer com esta dissertação é o estudo da influência do modelo de governança na adopção baseada em processos nas organizações portuguesas, nomeadamente a relação causa/efeito entre governança e gestão de processos aplicada às TI.

A utilização de boas práticas na gestão de serviço de TI, também é um tema que pretendemos abordar nesta dissertação, uma vez que o corpo de conhecimento existente é empírico e carece de prova.

⁶ - Esta fase decorreu em estreita colaboração com outros dois membros do grupo de investigação QUASAR, Jorge Freitas e José Carlos Costa, a desenvolverem igualmente as suas dissertações de mestrado sob orientação do Prof. Doutor Fernando Brito e Abreu.

Esta dissertação também me vai permitir fazer uma ponte entre a minha experiência profissional, baseada em projectos ITIL, com a investigação académica.

1.5 Organização da Dissertação

Para além do presente capítulo introdutório, esta dissertação é composta por um conjunto de sete capítulos.

- **Capítulo 2:** Este capítulo fornece uma panorâmica sobre o trabalho que tem sido desenvolvido nas áreas envolvidas a esta dissertação. O capítulo começa por descrever uma taxionomia que servirá para classificar as propostas apresentadas. Seguidamente, essas mesmas propostas são descritas e detalhadas. No final do capítulo é apresentada uma conclusão relativa ao estado da arte actual.
- **Capítulo 3:** Neste capítulo são apresentados alguns conceitos básicos sobre a governança de TI e qual a sua importância. Fala-se das principais decisões e arquétipos de TI, dos mecanismos de TI, dos perfis e abordagens da governança de TI e da forma como as empresas governam a TI. Finalmente, mostra-se como planear e avaliar a governança de TI e apresentam-se algumas recomendações para a governança de TI.
- **Capítulo 4:** Este capítulo relata detalhadamente a abordagem adoptada para determinar os arquétipos de governança de TI das empresas que fazem parte da amostra. É descrito o processo de elaboração, recolha e análise do questionário sobre a governança de TI. Fala-se das principais decisões e arquétipos de TI, dos perfis e abordagens da governança de TI e da forma como as empresas que fazem parte da amostra governam a TI.
- **Capítulo 5:** Este capítulo formaliza um conjunto de indicadores quantitativos que permitem exprimir a complexidade de um processo, com recurso à linguagem OCL, sobre um metamodelo BPMN.
- **Capítulo 6:** Neste capítulo são analisados os resultados obtidos nos capítulos 4 e 5 com recurso a uma análise de correlação entre os indicadores descritos no capítulo 5 e a um caso de estudo de uma empresa portuguesa.
- **Capítulo 7:** Neste capítulo são apresentadas as conclusões desta dissertação e é discutido, de que forma foram atingidas as contribuições previstas. Esta dissertação termina com a identificação das linhas de orientação gerais para a continuação futura do trabalho desenvolvido.
- **Anexo A:** Neste anexo é apresentado o inquérito de governança de TI realizado, os seus resultados e a interpretação dos mesmos.
- **Anexo B:** Este anexo mostra o metamodelo BPMN, o qual foi desenvolvido no âmbito de outra dissertação [Freitas, 2010] e as métricas OCL desenvolvidas para calcular a complexidade de um processo/modelo.

[Esta página foi intencionalmente deixada em branco]

Capítulo 2

Estado da Arte

Conteúdo

2 ESTADO DA ARTE.....	12
2.1 INTRODUÇÃO	12
2.2 DESCRIÇÃO DA TAXIONOMIA PROPOSTA	12
2.3 ENQUADRAMENTO DO TRABALHO RELACIONADO	15
2.3.1 <i>Governança de TI</i>	15
2.3.1.1 Avaliação do Estado da Arte.....	15
2.3.1.2 Revisão dos Artigos de Governança de TI	15
2.3.2 <i>Complexidade de Processos de TI</i>	23
2.3.2.1 <i>Processos de Gestão de Serviços de TI</i>	24
2.3.2.1.1 Avaliação do Estado da Arte	24
2.3.2.1.2 Revisão dos artigos Processos de Gestão de Serviços de TI	25
2.3.2.2 <i>Processos de Desenvolvimento</i>	30
2.3.2.2.1 Avaliação do Estado da Arte	30
2.3.2.2.2 Revisão dos Artigos Processos de Desenvolvimento	31
2.4 RESUMO DA TAXIONOMIA E DO ESTADO DA ARTE.....	36

Este capítulo fornece uma panorâmica sobre o trabalho que tem sido desenvolvido nas áreas domínio desta dissertação. O capítulo começa por descrever uma taxionomia que servirá para classificar as propostas apresentadas. Seguidamente, essas mesmas propostas são descritas e detalhadas. No final do capítulo é apresentada uma conclusão relativa ao estado da arte actual.

2 Estado da Arte

2.1 Introdução

Nesta secção descrevemos o trabalho relacionado relevante, classificado segundo uma taxionomia aqui proposta. O conjunto de artigos escolhidos foi seleccionado segundo 3 heurísticas: tempestividade, relevância e ajustamento ao tópico de interesse. Para este fim, pesquisámos palavras-chave sobre o tema desta dissertação em trabalhos actuais publicados por editoras de renome. O intuito foi escolher artigos que cruzassem a governança de TI com a complexidade dos processos e perceber se existe alguma ligação entre ambos. Com estas 3 heurísticas conseguimos restringir bastante a pesquisa e seleccionar o que realmente interessa. Após esta fase, definimos uma taxionomia para classificar os pontos fortes e fracos de cada artigo. Com esse objectivo e para melhor compreender, situar e comparar o trabalho relacionado, propomos uma taxionomia para os aspectos relacionados com a governança de TI e com a complexidade dos processos. Esta taxionomia inclui um conjunto de critérios que permitem seguir uma abordagem metódica, possibilitando o ganho de objectividade e comparabilidade neste estudo. A cada critério desta taxionomia está associada uma escala ordinal (sempre que possível) ou nominal, para ajudar a avaliar os pontos fortes e fracos de cada proposta contida nos artigos seleccionados.

A estrutura de cada proposta é constituída pelas seguintes secções: objectivo, resumo, crítica e classificação segundo a taxionomia proposta.

Na secção relativa ao **Objectivo**, apresenta-se uma breve descrição sobre a finalidade e o propósito do artigo/proposta; no **Resumo** descreve-se, em traços gerais, os principais detalhes e a importância do modelo de governança e/ou o modelo de maturidade de processos apresentado; na secção relativa à **Crítica**, apresenta-se um comentário construtivo face ao modelo apresentado, enumerando as suas vantagens e desvantagens; finalmente na secção **Classificação**, mostra-se uma avaliação baseada nos critérios definidos na taxionomia.

No fim de cada proposta será apresentado um quadro resumo com a classificação de todos os artigos analisados.

2.2 Descrição da Taxionomia Proposta

Durante a pesquisa dos trabalhos relacionados sobre modelos de governança de TI e modelos de maturidade dos processos, foram identificados alguns critérios específicos que orientaram a análise sobre o relacionamento entre os vários trabalhos em curso e o que será proposto nesta dissertação. Esses critérios tiveram em consideração a maneira de expressar e determinar os modelos de governança de TI e os modelos de maturidade dos processos.

1. Domínio da Aplicação

Com este critério pretendemos saber em que áreas de actuação é que os modelos em estudo têm sido usados com sucesso, na íntegra ou adaptados nos mais variados tipos de empresas.

Este critério pode ser classificado nas seguintes categorias, definidas na seguinte escala nominal:

- GTI – Governança de TI;
- GSS - Gestão de Software e Sistemas;
- GSTI - Gestão de Serviços de TI.

2. Aplicabilidade / Replicação

A questão que se coloca aqui é se a abordagem teórica do artigo em estudo pode ser utilizada na prática, ou seja, importa questionar se é possível aplicar a teoria à prática.

É fundamental escolhermos um modelo onde a sua implementação seja exequível. Por isso, temos que analisar bem a exposição dos processos descritos e conceitualizados pela teoria, e verificar se, de facto, podemos instanciar esse modelo teórico com exemplos da vida real.

Este critério pode ser caracterizado numa escala de avaliação ordinal de 4 categorias:

CC - Cobre completamente: Existe evidência de uma aproximação completa e metódica à prática definida no artigo em estudo.

CL - Cobre largamente: Existe uma clara evidência de aproximação metódica à prática definida no artigo em estudo, embora alguns aspectos não sejam cobertos.

CP - Cobre parcialmente: Existe uma reduzida evidência de aproximação à prática definida no artigo em estudo.

NC - Não cobre: Não existe evidência da aplicabilidade prática da proposta definida no artigo em estudo.

3. Formalização/Discriminação

Este critério pretende avaliar a forma como os modelos são apresentados. Pretendemos avaliar a capacidade de discriminação dos vários modelos propostos. Discriminar significa "fazer uma distinção". Questionámo-nos então se será possível distinguir os vários modelos/padrões apresentados no artigo em estudo.

Este critério pode ser caracterizado numa escala de avaliação ordinal de 4 categorias:

CC - Cobre completamente:

- Os conceitos introduzidos são clarificados e bem definidos;
- São apresentadas vantagens e desvantagens dos modelos apresentados;

- São apresentadas comparações entre os vários modelos/padrões propostos.

CL - Cobre largamente: Cobre 2 dos 3 tópicos apresentados em CC.

CP - Cobre parcialmente: Apenas cobre 1 dos 3 tópicos apresentados em CC.

NC - Não cobre: Não cobre nenhum dos tópicos apresentados em CC.

4. Diagnóstico

Este critério diz-nos se o artigo em estudo nos ajuda a conhecer ou a determinar o grau de maturidade de governança de TI ou dos processos de uma organização.

Este critério pode ser caracterizado numa escala de avaliação ordinal de 4 categorias:

CC - Cobre completamente:

- Fornece um conjunto de questões para avaliar a maturidade;
- Mostra evidências e trabalho experimental realizado numa avaliação de uma organização;
- Ensina, através de exemplos, domínios ou outros métodos/técnicas de como fazer uma avaliação à maturidade.

CL - Cobre largamente: Cobre 2 dos 3 tópicos apresentados em CC.

CP - Cobre parcialmente: Apenas cobre 1 dos 3 tópicos apresentados em CC.

NC - Não cobre: Não cobre nenhum dos tópicos apresentados em CC.

5. Validação

Este critério pretende verificar se algo cumpre com determinados critérios previamente estabelecidos. A validação deve estabelecer uma evidência documentada de que um processo é correcto e que cumpre com os parâmetros estabelecidos.

Este critério pode ser caracterizado numa escala de avaliação ordinal de 4 categorias:

CC - Cobre completamente:

- São apresentadas experiências, ensaios, provas, demonstrações ou casos de estudo reais;
- São identificadas ameaças à validade interna das propostas efectuadas, ou seja, aqui é interessante questionarmos se a interligação entre as variáveis dependentes e independentes são claras e inequívocas para fazer uma inferência causal (Leman, 1991). A validade interna está preocupada com conclusões onde uma variável independente é responsável pela variação na variável dependente. Questionamos, então, se o tratamento causa o efeito. Se a experiência conseguir estabelecer que o tratamento causa o efeito, então o estudo tem validade interna. Se, alguma coisa para lá do tratamento causar o efeito, então o estudo não tem validade interna;

- São identificadas ameaças externas à validade das propostas efectuadas, ou seja, aqui interessa questionarmos se é possível generalizar os resultados.

CL - Cobre largamente: Cobre 2 dos 3 tópicos apresentados em CC.

CP - Cobre parcialmente: Apenas cobre 1 dos 3 tópicos apresentados em CC.

NC - Não cobre: Não cobre nenhum dos tópicos apresentados em CC.

2.3 Enquadramento do Trabalho Relacionado

2.3.1 Governança de TI

2.3.1.1 *Avaliação do Estado da Arte*

A ligação entre os processos de negócios e a TI é muito forte, porém, não se pode considerar a gestão de TI como sendo a única ou a principal parte deste elo. Interessa por isso questionar como poderá ser maximizado o uso da TI para suportar as estratégias de negócios das empresas.

Pelas propostas apresentadas sobre governança de TI, observamos que, recentemente, são vários os estudos que estão a ser feitos para encontrar um referencial ideal de governança de TI. Os referenciais que estão disponíveis para apoiar a governança de TI são, em grande parte, limitados ao domínio da gestão (têm aspectos táticos mas pouco estratégicos). Até mesmo a única norma disponível para a governança de TI [International Standard ISO/IEC 38500, 2008] está, em grande medida, direccionada para lidar com questões de gestão, em vez de questões sobre governança de TI. Ainda pode demorar um pouco antes de um verdadeiro referencial de governança de TI ficar disponível [Bon, 2008].

Por todos os documentos analisados, foi possível encontrar abordagens inovadoras e importantes para o desenvolvimento de um referencial de governança de TI. Observou-se alguma convergência na utilização de elementos comuns nos vários artigos expostos. A governança tende a lidar com elementos organizacionais que são responsáveis pela tomada de decisões, de forma transparente. Assim, é fundamental termos bem definidos e clarificados os vários princípios, arquétipos e mecanismos de governança.

No entanto, ainda não está bem claro que a governança é essencialmente restrita ao fornecimento da infraestrutura para tomar estas decisões. O processo de tomada de decisão em si, não está incluído na governança de TI. Tomar decisões é geralmente aceite como sendo um dos aspectos de gestão, o qual está separado de governança. [Bon, 2008].

2.3.1.2 *Revisão dos Artigos de Governança de TI*

➤ **ISO/IEC 38500 – Modelo de governança de TI [International Standard, 2008]**

Objectivo: Esta norma estabelece princípios orientadores para dirigentes de organizações (incluindo proprietários, administradores, directores, sócios, executivos séniores, ou similares) sobre o uso

aceitável, efectivo e eficiente da TI dentro das suas organizações. Esta norma aplica-se à governança da gestão dos processos (e decisões) relativos à informação e serviços de comunicação utilizados por uma organização. Esta norma tem como intuito ajudar os dirigentes a equilibrarem os riscos e a incentivarem as oportunidades decorrentes da utilização de TI. Por fim, esta norma ainda estabelece um modelo e um vocabulário para a governança de TI.

Resumo: Esta norma define 6 princípios para a boa governança de TI:

- Estabelecer responsabilidades facilmente compreensíveis para a TI;
- Planear a TI para que ela ofereça o melhor suporte à organização;
- Validar as aquisições de TI;
- Sempre que necessário, garantir o melhor desempenho da TI;
- Garantir que a TI está em conformidade com as regras formais (definir as regras como orientação para pensar no lado da procura);
- Garantir que a utilização dos recursos de TI respeita os factores humanos.

Depois de delimitados estes 6 princípios, são definidos três conceitos:

- Avaliar: Os dirigentes devem examinar e fazer juízos sobre a utilização actual e futura das TIs, incluindo as estratégias, as propostas e os sistemas de fornecimento (sejam internos, externos, ou ambos). Na avaliação do uso de TI, os dirigentes devem considerar as pressões externas ou internas exercidas sobre os negócios, tais como as mudanças tecnológicas, as tendências económicas e sociais, e as influências políticas. Os dirigentes devem ainda realizar avaliações contínuas e ter em conta as actuais e futuras necessidades de negócio - os objectivos organizacionais actuais e futuros, que eles devem atingir, tais como manter vantagem competitiva, e os objectivos específicos das estratégias e das propostas que estão a avaliar;
- Dirigir: Os dirigentes devem atribuir responsabilidades e definir e implementar planos e políticas. Os planos devem definir o rumo para os investimentos em projectos de TI e operações de TI. As políticas devem estabelecer uma boa conduta no uso de TI. Os dirigentes devem assegurar que a transição de projectos para o estado operacional está devidamente planeada e gerida, tendo em conta impactos nos negócios e práticas operacionais, bem como sistemas de TI e de infraestrutura existentes. Os dirigentes devem ainda incentivar uma cultura de boa governança de TI nas suas organizações, exigindo gestores para fornecer informação atempada, para dar cumprimento à direcção e para estar em conformidade com os supracitados seis princípios da boa governança;
- Monitorizar: Os dirigentes devem monitorizar, através de adequados sistemas de medição, o desempenho da TI. Eles devem assegurar-se de que o desempenho está em conformidade com os planos, designadamente no que respeita aos objectivos do negócio. Os dirigentes também devem conferir que a TI está em conformidade com as obrigações externas (regulamentação, legislação, lei comum, contratual) e práticas de trabalho interna.

É a partir da conjugação dos princípios anteriormente descritos com estes três conceitos que se chega a um conjunto de orientações e indicações para a boa governança de TI.

Crítica: Existe apenas uma norma disponível para a governança de TI, que é a norma australiana AS8015:2005 [Standards Australia, 2005]. Baseada nesta norma australiana, a ISO/IEC 38500:2008 [ISO/IEC Standards, 2008] cobre aspectos relacionados com a governança de TI. Esta norma salienta as estruturas básicas da conformidade e do desempenho. No entanto, é muito sucinta em relação à especificação das questões organizacionais sobre governança de TI e, em vez disso, trata algumas questões simples sobre gestão.

Classificação:

Critérios	International Standard 2008
Domínio da Aplicação	GTI
Aplicabilidade / Replicação	CP
Diagnóstico	NC
Formalização / Discriminação	CP
Validação	NC

Tabela 2.1: Classificação baseada nos critérios definidos na taxionomia

➤ **This is NOT IT Governance [Bon, 2008]**

Objectivo: Compreender o que é, e o que não é, a governança de TI. Introduzir uma disciplina de apoio a processos de negócio - “*Information Support*”. Gerir um domínio de acordo com o “*Principle of Separation of Concerns*” [Piessens, 2002]. Aplicar o modelo SAME (*Strategic Alignment Model Enhanced*) [Bon and Hoving, 2007].

Resumo: O modelo SAME pode ser usado como o "padrão básico" para questões de gestão de processos de negócio (“*Information Support*”) nas organizações. Este modelo descreve a responsabilidade e os elementos do processo, e pode, ainda, ser utilizado para resolver questões organizacionais, por exemplo organizar o *Service Desk*.

Segundo o modelo SAME, “*Information Support*” fornece-nos dois domínios de responsabilidade distintos: “*Information Management*” (IM), onde sistemas de suporte de informação são concebidos e controlados, e “*Information Technology*” (IT), onde os sistemas de informação são construídos e testados. Combinando estes subdomínios, obtemos um modelo onde podemos expressar a gestão de negócio, a gestão de informação e a gestão da tecnologia numa vertente estratégica, tática e operacional:

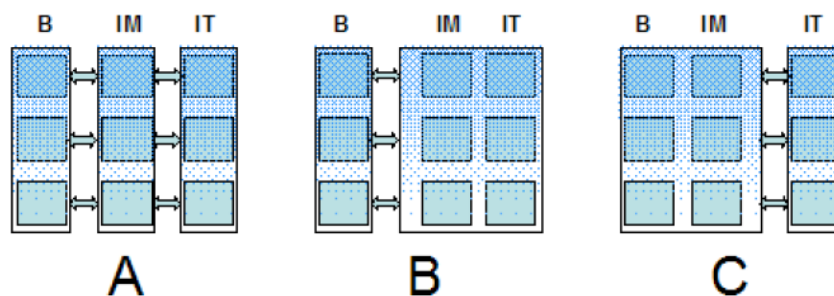


Figura 2.4: A posição do domínio de Gestão de Informação entre o Negócio e a Tecnologia de Informação no modelo SAME (extraído de [Bon, 2008]).

A) IM posiciona-se equidistante do domínio do Negócio e da TI, em muitos casos é emblemático para as organizações que tentam implementar a IM como uma função de ligação; B) IM posiciona-se como uma extensão da função de TI. As responsabilidades da IM são largamente delegadas ao domínio da Tecnologia, onde os serviços de TI são produzidos; C) IM posiciona-se como uma extensão da função de Negócio. Aqui, a informação é considerada como um bem do negócio, e a relação com a tecnologia pode ser contratual: a TI é uma função de apoio, a ser gerida como tal, e, possivelmente, governada através de *outsourcing*. As responsabilidades da IM e do Negócio são fortemente ligadas e a TI pode ser considerada como uma mercadoria substituível, a ser fornecida por qualquer fornecedor adequado. Isto evidencia a questão de se saber onde se posiciona, na organização, a responsabilidade para IM e para a TI. Esta questão, normalmente é uma questão de governança de TI. Resumidamente, este artigo centra-se na questão do posicionamento do domínio IM.

Crítica: Embora esta abordagem apresente um modelo de governança de TI, a delimitação do domínio da gestão está presente. A governança e a gestão de TI são duas entidades separadas. Existe uma clara distinção entre governança e gestão, sugerindo que a governança permite a criação de um modelo no qual outros conseguem gerir eficazmente as suas funções. Este modelo SAME de Jan Van Bon é muito bom para representar os referenciais a nível estratégico, tático e operacional. Mesmo sabendo o domínio em que se insere a gestão de informação, isto não é suficiente para termos um grau de maturidade de governança de TI elevado. Daí, termos considerado este artigo incompleto. Este modelo pode ser usado como um complemento na tomada de decisões de TI.

Classificação:

Critérios	Jan Van Bon 2008
Domínio da Aplicação	GTI
Aplicabilidade / Replicação	CL
Diagnóstico	NC
Formalização/ Discriminação	CL
Validação	CP

Tabela 2.2: Classificação baseada nos critérios definidos na taxionomia

Os próximos 4 artigos são resultado de pesquisas do CISR⁷ (*Center for Information Systems Research*) da *Sloan School of Management* do MIT (*Massachusetts Institute of Technology*). Estes 4 artigos tiveram a participação de *Peter Weill*⁸, cientista sénior e presidente do CISR. O trabalho de *Weill* centra-se no papel, no valor e na governança de TI nas empresas. O seu contributo nesta área tem sido fundamental, nomeadamente, na especificação de um modelo que determina os direitos e responsabilidades de decisão, para encorajar um comportamento desejável no uso da TI.

⁷ CISR - a research group at the MIT Sloan School of Management <http://mitsloan.mit.edu/cisr/>

⁸ Peter Weill – bibliografia em http://mitsloan.mit.edu/faculty/detail.php?in_spseqno=15571&co_list=F

➤ **Banknorth: Designing IT Governance for a Growth-Oriented Business Environment**
[Hoffman and Weill, 2007]

Objectivo: Artigo sobre um caso de estudo. É feita uma descrição exaustiva de uma abordagem de um banco na Inglaterra sobre a implementação de um referencial de governança de TI. Este estudo foi baseado no livro de Peter Weill e Jeanne Ross [Weill and Ross, 2004].

Resumo: Este caso de estudo descreve os desafios da implementação de governança de TI num banco regional da Inglaterra, o “*Banknorth*”, que estava a atravessar uma fase de crescimento explosivo e muito rentável, através de fusões e aquisições. Passou de um banco local a um banco regional. Foi feito um levantamento de todos os processos de negócio, da cultura e da estratégia adoptada. No novo referencial de governança de TI, foram decididos quais os mecanismos para os diferentes domínios de decisão e como deveria ser feita a comunicação entre eles. Seis meses após a implementação, o CIO avaliou a eficácia do novo referencial de governança de TI convidando uma equipa de estudantes da MIT *Sloan School of Management*. Esta equipa reuniu informação através de um inquérito da MIT e apresentou os resultados. O projecto de governança teve notas altas, mas também foram identificados alguns aspectos negativos. O *Banknorth* teve em conta os resultados da avaliação nas suas iniciativas de melhoria do seu modelo de governança de TI.

Crítica: Este documento prova como é possível implementar o referencial desenhado por *Peter Weill* e seus colaboradores. *Weill* foi inspirado pelo Professor *Thomas H. Davenport* que escreveu, em 1997, num dos seus livros, sobre os arquétipos de governança de TI [Davenport, 1997]. Este artigo é muito rico, na medida em que apresenta as questões e respostas feitas pelos estudantes da MIT aos colaboradores do *Banknorth*, mostra como foram feitos os cálculos para avaliar o desempenho da governança de TI e ilustra os resultados com gráficos. Contudo, a maneira como este artigo descreve todo o processo de implementação é um tanto superficial, na medida em que não explica as técnicas e os conceitos em que se baseou.

Classificação:

Critérios	Hoffman and Weill 2007
Domínio da Aplicação	GTI
Aplicabilidade / Replicação	CC
Diagnóstico	CC
Formalização/ Discriminação	NC
Validação	CL

Tabela 2.3: Classificação baseada nos critérios definidos na taxionomia

➤ **IT Governance on One Page [Weill and Ross, 2004]**

Objetivo: Artigo sobre um referencial de governança de TI, que incluiu um ou mais projectos de pesquisa da CISR (*Center for Information Systems Research*). Este trabalho oferece uma avaliação e um referencial para ajudar as empresas na concepção e na comunicação da governança de TI.

Resumo: Este artigo descreve um referencial de governança de TI.

Começa por mostrar como fazer uma rápida avaliação à governança de TI de uma empresa e calcular o seu desempenho. Para avaliar e comparar a governança de TI, os autores propõem uma matriz que representa como é que as decisões podem ser tomadas. Descrevem os cinco grandes domínios de TI e os diferentes arquétipos que podem ser utilizados para cada um desses domínios. Uma matriz completa de governança mapeia os tipos de decisões e os arquétipos para a tomada dessas decisões. Para sabermos como é que essas decisões serão tomadas e monitorizadas, temos que implementar um conjunto de mecanismos de governança.

Este artigo ainda descreve três abordagens da governança de TI: centralizada, descentralizada e híbrida.

Crítica: Este artigo apresenta concepções interessantes, bem como apresenta um referencial de governança. Porém, os conceitos introduzidos não são descritos exhaustivamente, ou seja, os princípios, os arquétipos, os mecanismos de governança são brevemente descritos, fazendo com que, por vezes, não consigamos distingui-los. É apresentado um conjunto de questões para a avaliação do desempenho e para as decisões de TI. Contudo, não é clarificado como deverá ser realizado o levantamento dos arquétipos e dos mecanismos de governança. São apresentados casos práticos para provarem a sua validação, mas o autor não mostra, nem explica, o que limita esta proposta de referencial.

Classificação:

Critérios	Weill and Ross 2004
Domínio da Aplicação	GTI
Aplicabilidade / Replicação	CL
Diagnóstico	CP
Formalização/ Discriminação	CP
Validação	CP

Tabela 2.4: Classificação baseada nos critérios definidos na taxionomia

➤ **Don't just Lead, Govern: How Top-Performing Firms Govern IT [Weill, 2004]**

Objetivo: Artigo sobre governança de TI que incluiu um ou mais projectos de pesquisa da CISR (*Center for Information Systems Research*). Com base nas melhores práticas de 300 empresas em 23 países, este artigo fala sobre a variedade de perfis de governança de TI. As empresas atribuem “direitos de decisão” a diferentes “arquétipos” para governar os cinco fundamentais domínios de decisão de TI. Um caso de estudo da “*State Street Corporation*” (líder mundial fornecedor de serviços

financeiros a investidores institucionais) ilustra a evolução da governança de TI e um método para representar esquematicamente a governança de TI.

Resumo: Este artigo mostra como os gestores de topo governam a TI de forma diferente uns dos outros e de forma diferente da média das empresas. As empresas líderes em crescimento descentralizam mais os seus direitos de decisão de TI, colocando as capacidades da TI nas unidades de negócio. As empresas líderes no lucro, centralizam mais a tomada de decisão e os líderes de negócio seniores é que tomam as grandes decisões de TI. Os gestores de topo concebem a sua governança de TI para reforçar os seus objectivos de desempenho e vincular a governança de TI à governança dos seus outros activos importantes na empresa e comportamentos desejados.

Em suma, este artigo expõe como é que os gestores de topo governam e menciona os factores críticos de sucesso da governança de TI.

Crítica: Este artigo é bastante detalhado em relação aos conceitos descritos. Clarifica os princípios e arquétipos que caracterizam a governança de TI e tem uma nota introdutória onde explica porque é que a governança de TI é importante. Tanto os arquétipos, como os direitos de decisão são exemplificados com as várias empresas que entraram neste estudo, mostrando a percentagem de empresas que opta pelos vários caminhos possíveis. Este artigo mostra o que existe no mercado, mostra como os gestores de topo governam as suas empresas, mas não mostra como implementar estas medidas de governança.

Classificação:

Critérios	Weill 2004
Domínio da Aplicação	GTI
Aplicabilidade / Replicação	CP
Diagnóstico	NC
Formalização/ Discriminação	CC
Validação	CP

Tabela 2.5: Classificação baseada nos critérios definidos na taxionomia

➤ Don't just Lead, Govern: Implementing Effective IT Governance [Weill and Woodham, 2002]

Objectivo: Artigo sobre governança de TI que incluiu um ou mais projectos de pesquisa da CISR (*Center for Information Systems Research*). É objectivo neste artigo definir governança de TI, enumerar os requisitos de uma governança de TI efectiva e propor um referencial de governança de TI, que especifique como é que as decisões são tomadas nos principais domínios de TI.

Resumo: Muitas empresas estão a criar estruturas de governança de TI que incentivem o comportamento e conduzam à concretização dos objectivos de desempenho de negócio da empresa. A governança eficaz de TI exige uma análise cuidadosa sobre quem toma as decisões e como é que as

decisões são tomadas em, pelo menos, quatro domínios de TI críticos: princípios, infraestrutura, arquitectura e investimentos e prioritizações. Foram feitos vários estudos em unidades de negócio de grandes empresas, nos Estados Unidos e na Europa, e chegou-se à conclusão que as empresas típicas governam a TI seguindo, geralmente, linhas orientadoras onde os direitos de decisão são rigorosamente controlados. Para ajudar a compreender e conceber uma governança mais eficaz, os autores propuseram um referencial de governança TI. Este referencial harmoniza arquétipos de governança desejados com um conjunto de mecanismos de governança. O referencial é ilustrado com a governança eficaz de TI na “*State Street Corporation*”.

Crítica: É descrito como é que as empresas típicas governam a TI, e os vários pontos de *stress* e desafios que os padrões de governança de TI têm nestas empresas. Infelizmente, temos muito pouca informação sobre os conceitos utilizados neste artigo. Não são apresentados os critérios em que se basearam para escolher os domínios ou mecanismos supracitados. Também não é explicado o que é que cada um deles quer dizer ao certo. É feita uma comparação com as empresas governadas por gestores de topo. O referencial de governança de TI foi desenvolvido segundo três visões: (1) objectivos de negócio – comportamento desejável, (2) estilos de governança de TI – mecanismos de governança de TI e (3) objectivos de desempenho do negócio – métricas. Estudou-se este referencial aplicando o caso da “*State Street*”. Contudo, este artigo não descreve formas de avaliação, isto é, não sabemos como é que foi feito o levantamento neste caso de estudo. Também não se clarifica em que é que este referencial nos ajuda, nem as mais-valias que traz para a organização. As suas vulnerabilidades também não são apresentadas.

Classificação:

Critérios	Weill and Woodham 2002
Domínio da Aplicação	GTI
Aplicabilidade / Replicação	CL
Diagnóstico	NC
Formalização/ Discriminação	CP
Validação	CP

Tabela 2.6: Classificação baseada nos critérios definidos na taxionomia

➤ The IT organization modeling and assessment tool: Correlating IT governance maturity with the effect of IT [Simonsson and Johnson, 2008]

Objectivo: Este artigo apresenta uma ferramenta de avaliação e modelação da organização de TI (ITOMAT - *IT Organization Modeling and Assessment Tool*). Esta ferramenta foi criada para superar fraquezas de operacionalização e de subjectividade no referencial CobiT.

Resumo: Neste artigo, é proposto um método para avaliar o grau de maturidade da governança de TI dentro de uma organização. A ferramenta ITOMAT foi aplicada para avaliar a maturidade da governança de TI em 4 casos de estudo. Simultaneamente, foram recolhidas métricas externas do

efeito da TI e correlacionadas com os níveis de maturidade. Baseado nestas correlações, foi criado um modelo que interliga as medidas internas e externas. Este modelo pode ser usado para prever o efeito da TI, dando o nível de maturidade dos processos de TI.

Crítica: Este artigo está bastante próximo da nossa dissertação, pois a questão que se coloca aqui é se a boa governança de TI melhora o efeito da TI, mas é fortemente baseado em CobiT. A nossa dissertação vai ser baseada em ITIL. Os autores propuseram a ITOMAT, que foi baseada no referencial CobiT, aproveitando os seus benefícios e mitigando algumas das suas fraquezas. Este método está muito bem conseguido pois contém duas partes: a linguagem de modelação, que fornece uma representação descritiva de como a TI é governada dentro da organização avaliada, e a análise do referencial que fornece suporte para avaliar se a estrutura de governança actual é boa ou má. Este tipo de avaliação é essencial para a boa monitorização e para a gestão dos processos e da estrutura de TI. Esta ferramenta supera o problema da eficácia, da fiabilidade e também do custo que são tão comumente associadas com os métodos de hoje. Contudo, neste artigo não são definidos arquétipos/perfis de governança de TI, como nós fizemos nesta dissertação.

Classificação:

Critérios	Simonsson and Johnson, 2008
Domínio da Aplicação	GTI e GSTI
Aplicabilidade / Replicação	CL
Diagnóstico	CL
Formalização/ Discriminação	CP
Validação	CL

Tabela 2.7: Classificação baseada nos critérios definidos na taxionomia

2.3.2 Maturidade de Processos de TI

Estes artigos têm em comum o facto de descreverem os modelos de qualidade mais utilizados actualmente, bem como modelos de processos, guias ou formulários que podem ser utilizados directamente como uma base para a implementação do modelo desejado.

As propostas apresentadas estão divididas em dois grupos:

- Modelos de Qualidade de Serviços de TI, onde se apresentaram artigos sobre ITIL e CobiT (secção 2.3.2.1 – Processos de Gestão de Serviços de TI);
- Modelos de Qualidade de Sistemas e Software, onde se apresentaram artigos sobre o CMM, CMMI e o SPICE (secção 2.3.2.2 – Processos de Desenvolvimento).

2.3.2.1 Processos de Gestão de Serviços de TI

2.3.2.1.1 Avaliação do Estado da Arte

O ITIL é um guia de referência que conseguiu, de forma bastante simples e clara, segmentar os processos de gestão do ambiente de TI e estabelecer pontos de controlo sobre cada uma das suas etapas. A grande vantagem é que os processos podem ser implementados por etapas, facilitando toda a cultura gradual de uma organização. A grande desvantagem é não ter as chamadas “*quick wins*”⁹ e ser um processo muito moroso. O ITIL sobrepõe-se ao CMM em determinadas áreas, como por exemplo na gestão de configuração. O ITIL investiga, por exemplo, as mudanças efectuadas nos sistemas operacionais, mas a qualidade dessas mudanças em termos do número e da gravidade de problemas resultantes delas é uma métrica do CMM. No ITIL, existe uma vasta colecção de *Key Performance Indicators* (KPIs) que apoiam a introdução de um referencial compreensivo para controlar os processos. Isto inclui avaliações periódicas da qualidade dos processos de Gestão de Serviços de TI. Os KPIs são utilizados para avaliar se os processos de uma organização de TI - os processos ITIL - estão a ser executados de acordo com as expectativas. Definir KPIs adequados é, acima de tudo, decidir sobre o que exactamente é considerado uma implementação bem sucedida de um processo. Depois de definirmos os KPIs, torna-se possível determinar e medir indicadores específicos. Assim, já temos condições de avaliar a qualidade dos processos, que por sua vez é a base para a contínua optimização e aperfeiçoamento da concepção do processo. A selecção dos KPIs adequados irá, entre outras coisas, depender da possibilidade de realmente medir os indicadores. Os KPIs e os correspondentes procedimentos de medição são, portanto, um importante contributo para o sistema de requisitos.

O CobiT procura ocupar o espaço entre a Gestão de Riscos, voltada para o negócio, a Gestão de Serviços em TI e a Gestão da Segurança da Informação. Estes modelos de gestão compõem-se de boas práticas específicas, segundo a sua área principal, e possuem funções complementares. Desta forma, o CobiT permite alinhar os objectivos dessas áreas de conhecimento às estratégias e princípios de governança corporativa, garantindo assim, que os processos e actividades desempenhadas pelas respectivas áreas e funções corporativas concorram de forma sistemática para o alcance dos objectivos de negócio e para a redução dos riscos operacionais. Para estruturar os processos de TI, usando o referencial do CobiT, será necessário avaliar os processos existentes em TI na organização, fazer uma análise de diagnóstico e risco em relação ao CobiT e modificar o que for necessário para atender aos objectivos de controlo, fazendo uso dos conceitos, metodologias e ferramentas disponíveis no mercado. Isso deve ser feito com base numa metodologia para a modelação de processos, caso contrário as representações e as formas de abordagens podem diferir muito entre as várias áreas em TI, dificultando a integração das mesmas. Independente da metodologia adoptada, e conforme a fase da modelação dos processos de TI, a estrutura de controlos do CobiT auxilia na estruturação do

⁹ Vitórias rápidas que apoiarão e darão visibilidade do sucesso das iniciativas de ITIL

levantamento, compreensão, análise de diagnóstico e risco, acompanhamento da implementação e monitorização dos processos implementados. O CobiT ajuda uma organização a identificar o nível de maturidade actual e como evoluir para melhorar os seus processos.

Assim, o CobiT é mais aconselhado sempre que o objectivo seja fazer medições ou auditar. Sempre que o objectivo seja melhorar os processos em termos de qualidade e redução de custos o ITIL é mais aconselhável, tal como observou Ricardo Mansur em [Mansur, 2007].

Podemos concluir então, que os modelos de qualidade, dizem “o que” e não “como” fazer. Determinar este “como” é um trabalho adicional bastante grande que as organizações devem realizar aquando da adopção destas normas.

2.3.2.1.2 *Revisão dos artigos Processos de Gestão de Serviços de TI*

➤ **An ISMS (Im) – Maturity Capability Model [Woodhouse, 2008]**

Objectivo: Este trabalho propõe um modelo de maturidade de processos único para avaliar a capacidade e maturidade dos processos que afectam o Sistema de Gestão de Segurança de Informação (ISMS - *Information Security Management System*) dentro de uma organização.

Resumo: Os modelos de maturidade têm sido utilizados para avaliar e orientar iniciativas de melhoria de processos, desde desenvolvimento de software até engenharia de sistemas, aquisição de produtos, gestão de equipa ou segurança da informação, apenas para citar algumas. Estes modelos são baseados na melhoria do processo e fornecem um referencial para orientar e medir a implementação e a melhoria dos processos. Em todos estes modelos, quanto maior for o nível de avaliação de uma organização, melhor (na teoria) a organização está a definir, avaliar e melhorar a sua capacidade dos processos.

O modelo proposto neste artigo descreve nove níveis de maturidade de processo, quatro dos quais são abaixo dos cinco níveis definidos actualmente nos modelos mais populares.

Crítica: Este artigo é bastante inovador ao descrever um novo modelo que fornece uma indicação do estado actual dos processos de uma organização e fornece uma orientação de como os processos são definidos, implementados e melhorados através dos estágios de maturidade. A intenção do modelo é destacar o papel que a cultura corporativa desempenha, ao garantir os activos de informação da organização e ao permitir uma melhor avaliação do seu verdadeiro nível de maturidade.

Contudo, este artigo é bastante teórico. Não existem evidências, nem trabalho experimental, que demonstre que esta abordagem é válida na realidade do mundo empresarial. Além disso, este artigo apenas descreve as várias fases do modelo proposto, não exemplificando como implementá-lo.

Classificação:

Crítérios	Woodhouse, 2008
Domínio da Aplicação	GSTI
Aplicabilidade / Replicação	CP
Diagnóstico	NC
Formalização/ Discriminação	CP
Validação	NC

Tabela 2.8: Classificação baseada nos critérios definidos na taxionomia

➤ **IT Business Process Performance Management: Case Study of ITIL Implementation in Finance Service Industry [Spremic et al, 2008]**

Objectivo: Este artigo teve como finalidade apresentar um caso de estudo sobre a gestão de serviços de TI, na indústria financeira, onde foram implementadas sugestões baseadas no conjunto de boas práticas ITIL. Descrevem-se os vários problemas que surgiram durante a sua implementação e revelam-se os resultados obtidos expressos através de vários indicadores chave de desempenho (KPIs - *Key Performance Indicators*). KPI é um termo utilizado para descrever os valores fundamentais e relações relevantes para um processo de negócio.

Resumo: Organizações bem sucedidas compreendem, gerem e medem sistematicamente a qualidade do desempenho da TI. Nos últimos anos, foram propostos vários referenciais de melhores práticas, nas áreas de gestão de processos de TI, tais como o ITIL e o CobiT, os quais ajudam a gestão a medir o desempenho da TI.

Este artigo apresenta um caso de estudo de uma empresa que opera na Croácia em mais de 100 locais e com 5000 funcionários. Esta empresa oferece serviços financeiros para bancos comerciais croatas. A fim de aumentar a qualidade dos seus serviços, a empresa reconheceu a necessidade de instituir processos de TI adequados e ferramentas para a automação dos seus processos. Também reconheceu a necessidade de formar os seus funcionários, no sentido de estabelecer atitudes correctas e competências que deverão permitir apoiar os serviços de TI proactivamente. A empresa percebeu que, se aumentasse a satisfação do cliente, iria obter melhor reputação no mercado concorrencial. Assim, a empresa acreditou que a adopção do referencial ITIL poderia ajudar a melhorar os seus processos. Durante a sua implementação, a empresa debateu-se com alguns problemas, como por exemplo:

- o novo sistema teve que ser aceite pelo quadro da administração da empresa. Devido à baixa posição na hierarquia da organização, o pessoal de TI tinha que ouvir os seus superiores hierárquicos, em vez de seguir os novos processos de ITIL implementados;
- a implementação gera uma grande quantidade de novos documentos e manuais que ameaçam burocratizar a organização;
- também houve alguns problemas causados pelos parceiros de *outsourcing* e pelos fornecedores de software, especialmente na área de automação de processos ITIL.

A fim de obter resultados a partir da implementação ITIL e obter melhorias em procedimentos de TI alterados e processos de TI, a empresa sentiu a necessidade de definir (KPIs). Para cada processo, os KPIs representam um objectivo que a empresa precisava de alcançar. Após terem estabelecidos quais os KPIs necessários para monitorizar as melhorias, para cada processo foi medido o desempenho antes da implementação de ITIL, a fim de compará-lo com os resultados após a sua implementação. Os resultados mostraram, sem dúvida, como a empresa melhorou os seus principais processos de negócio após a implementação ITIL.

Crítica: Este artigo mostra-nos como um referencial de boas práticas, através de KPIs, consegue mostrar as melhorias alcançadas por uma organização. Esta técnica não tem uma escala que nos possa situar sobre a nossa posição a nível de maturidade de processos. É possível acompanhar, medir e verificar se um dado processo está a evoluir.

Uma das grandes vantagens para as empresas implementarem ITIL é que os seus processos têm associado um conjunto muito vasto de indicadores que permitem uma avaliação mais rigorosa da sua implementação. Isto é fundamental para as empresas se auto-avaliarem e poderem empreender o processo de melhoria contínua que está subjacente ao ITIL.

Contudo, deve-se ter em consideração que o ITIL não descreve como é que os seus processos devem ser implementados. O ITIL apenas descreve os objectivos e os KPIs a atingir, oferecendo um *roadmap* às empresas para avaliarem o sucesso da sua implementação. Em termos práticos, as empresas terão que determinar quais são os seus pontos fortes, e fracos, para uma implementação ITIL com sucesso e, consequentemente avaliar o esforço que, será necessário em termos de recursos humanos internos e externos, bem como a formação que deverão ter.

Classificação:

Critérios	Spremic, Zmirak, Kraljevic, 2008
Domínio da Aplicação	GSTI
Aplicabilidade / Replicação	CL
Diagnóstico	NC
Formalização/ Discriminação	CP
Validação	CL

Tabela 2.9: Classificação baseada nos critérios definidos na taxionomia

➤ A Metrics Generation Model for Measuring the Control Objectives of Information System Audit [Nicho and Cusack, 2007]

Objectivo: Neste artigo é proposto um modelo para gerar métricas que representem os indicadores chave de desempenho (KPI's) e os objectivos de controlo do CobiT, aplicando o modelo GQM (*Goal, Question, Metric*) [Solingen, 1999].

Resumo: Alguns dos factores que fazem com que as empresas dêem mais importância à gestão, ao controlo e à medição dos sistemas de informação devem-se ao colapso da *Enron Inc.* [Wikipedia, *Enron_scandal*], à necessidade de um sistema de comunicação e divulgação financeiro melhor, ao risco associado à informação, aos investimentos feitos pelas empresas em recursos de TI ou à necessidade de se ser competitivo no mercado. Todos esses factores ressaltam a necessidade de medir o desempenho ou a eficácia dos sistemas de informação. O objectivo deste trabalho é o de transpor o modelo GQM em CobiT, adoptando os objectivos de controlo detalhados do CobiT e seguindo as orientações do GQM para a geração de métricas que permitam medir o processo e as entidades que são auditadas durante um exercício de auditoria de TI. Este modelo está dividido em 3 níveis: nível conceptual (objectivo), nível operacional (questões) e nível quantitativo (métricas). O CobiT cobre 4 domínios: (i) planear e organizar, (ii) adquirir e implementar, (iii) entregar e dar suporte e, (iv) monitorizar e avaliar. Estes domínios possuem 34 processos e 2 objectivos de controlo para cada processo. O CobiT incide em métricas, nomeadamente KPI's e KGI's (Key Goal Indicators) (ambos indicadores-chave de sucesso). As métricas são classificadas como KGI quando estamos a falar dos objectivos da empresa, dos objectivos e dos processos da TI e dos objectivos das actividades. Quando falamos de quão bem o processo está a ser desempenhado, classificamos a métrica como KPI. A abordagem GQM [Nicho and Cusack, 2007] é "uma forma sistemática, para adequar e integrar os objectivos de uma organização em objectivos mensuráveis e refiná-los em valores mensuráveis". Isto implica que este modelo pode ser utilizado pelo referencial CobiT, traduzindo os seus 34 processos de nível elevado e os seus objectivos de nível baixo em valores mensuráveis. O modelo também aborda o aspecto "como" da implementação, uma vez que não só ajuda na formulação de um programa de medição eficaz, mas também orienta na sua implementação.

Crítica: Enquanto que o CobiT é um referencial de auditoria que tem vindo a evoluir ao longo dos últimos oito a dez anos, o GQM é um modelo para gerar métricas que meçam os diversos objectivos relacionados com aspectos do processo de desenvolvimento de software (Engenharia de Software). A integração destes dois modelos, descrita neste artigo, é apenas uma demonstração teórica da geração de métricas usando o modelo GQM num referencial de auditoria. Lá por um modelo ser bem sucedido num contexto (Engenharia de Software), isso não significa que ele será bem sucedido noutro contexto (medição em auditorias de TI), a menos que o modelo teórico seja submetido a testes empíricos num ambiente do mundo real. A aplicação do GQM noutras áreas que incluem diversos processos, produtos, entidades e actividades exige o refinamento do referencial, especialmente o *template* para desenvolver as questões.

Teoricamente, este artigo demonstrou que é possível construir um sistema de medição. Este é um passo para a criação de registos que guardem o estado das entidades, actividades e processos de um sistema de informação de uma organização. O modelo apresentado neste artigo é vantajoso na medida em que pode dar mais informações específicas e quantitativas sobre o estado actual dos sistemas de informação de uma organização. A informação pode ser utilizada para melhorar a estrutura existente

dos sistemas de informação para a realização dos objectivos de qualidade e, o referencial pode fornecer transparência nas operações dos sistemas de informação devido a um melhor controlo e monitorização.

Em conclusão, este artigo propõe um modelo para medir a maturidade dos processos de uma empresa, mas não explica como implementá-lo, talvez porque ainda não tenha sido testado e, consequentemente, validado.

Classificação:

Crítérios	Nicho and Cusack, 2007
Domínio da Aplicação	GSTI
Aplicabilidade / Replicação	CP
Diagnóstico	NC
Formalização/ Discriminação	CL
Validação	NC

Tabela 2.10: Classificação baseada nos critérios definidos na taxionomia

➤ Re-engineering IT Internal Controls: Applying Capability Maturity Models to the Evaluation of IT Controls [Debreceeny, 2006]

Objectivo: Este artigo relata uma tentativa exploratória para determinar a capacidade e maturidade das organizações. Foram abordadas questões sobre métricas na medição da capacidade e da maturidade e na sua correlação com o estado do controlo interno sobre os relatórios financeiros.

Resumo: Este trabalho introduziu uma nova metodologia para avaliar a capacidade e maturidade de uma entidade em cada um dos elementos do referencial CobiT. Uma componente importante, entre as várias ferramentas e elementos que compõem o referencial CobiT é o *Capability Maturity Model* (CMM) [Capability Maturity Model, 2008], que está incluído nas orientações de gestão do CobiT. Estas orientações também incluem factores de sucesso críticos (CSFs), *Key Performance Indicators* (KPIs) e *Key Goal Indicators* (KGIs). Uma importante característica distintiva do CMM é a sua abordagem sustentada na engenharia de desenvolvimento de processos de controlo. Isto permite-lhe ter técnicas suficientemente genéricas para serem utilizadas numa variedade de processos de engenharia. Assim, este modelo permite que um determinado nível do CMM possibilite, aos gestores, determinarem a capacidade de uma entidade. Estes conseguem, assim, gerir os seus riscos e resultados estratégicos e operacionais, para um processo específico. Um certo número de premissas foram desenvolvidas para cada objectivo de controlo, de modo a permitirem a auto-avaliação da capacidade e da maturidade.

Crítica: O CobiT é constituído por 4 domínios. Cada domínio tem um conjunto de objectivos de controlos. Neste artigo são tratados vários assuntos baseados nestes objectivos de controlo, de modo a obter-se uma avaliação quantitativa da capacidade e maturidade de uma dada organização. O nível de maturidade foi calculado de acordo com as respostas dadas. Cada resposta tinha um peso aplicado:

“De modo algum” (0), “Um pouco” (1/3), “Até um certo nível” (2/3) e “Completamente” (1). Por fim, a cada uma das respostas ponderadas foi aplicado o nível apropriado CMM, desde 1.0 (Inicial / *Ad-hoc*) até 5.0 (Otimizado).

Este modelo é bastante simples, mas tem algumas limitações. Primeiro, só existem dados de uma organização. Antes de podermos concluir e tirar ilações deste modelo, temos que sondar um número significativo de organizações. Depois, também não foi observado se um determinado nível de capacidade e de maturidade correlaciona com um dado nível de controlo interno. Não há estudos que nos permitam compreender esta relação. Existem questões interessantes que surgem, como por exemplo, quais são as influências da escolha da concepção da gestão de TI, tais como o *in-sourcing* versus *out-sourcing*, o nível de desenvolvimento interno versus o nível de aquisição de pacotes de software, a importância da TI para os processos que acrescentam valor à entidade, nos níveis de capacidade e maturidade.

Classificação:

Critérios	Debreceeny, 2006
Domínio da Aplicação	GSS e GSTI
Aplicabilidade / Replicação	CP
Diagnóstico	NC
Formalização/ Discriminação	CL
Validação	CL

Tabela 2.11: Classificação baseada nos critérios definidos na taxionomia

2.3.2.2 Processos de Desenvolvimento

2.3.2.2.1 Avaliação do Estado da Arte

O CMMI (*Capability Maturity Model Integration*) é o mais recente modelo de maturidade para desenvolvimento de software do SEI. Derivado do modelo CMM, o CMMI surgiu da percepção de que o software básico e as aplicações são desenvolvidos em contextos integrados.

O CMMI organiza as práticas, que já são consideradas efectivas, numa estrutura que visa auxiliar a organização a estabelecer prioridades para melhoria e também fornece um guia para a implementação dessas melhorias. O primeiro passo a ser dado é a identificação do nível de maturidade em que a empresa se encontra actualmente; uma vez que este denota o nível de maturidade a ser alcançado pelas empresas, visando ajudá-las no desenvolvimento e manutenção dos projectos de software, como também melhorar a capacidade dos seus processos. Após a verificação do nível de maturidade da empresa, verifica-se qual a próxima etapa a ser alcançada e quais as competências que devem ser adquiridas neste processo. Esta fase é importante, pois permite alcançar o sucesso e, consequentemente, haverá uma melhoria na qualidade dos serviços e produtos fornecidos pela área de tecnologia das organizações.

A ISO/IEC 15504, também conhecida como SPICE, é a norma que define o processo de desenvolvimento de software. O SPICE possui níveis de capacidade para cada processo assim como o CMMI. O SPICE define um modelo de referência de processo que identifica e descreve um conjunto de processos considerados universais e fundamentais para a boa prática da Engenharia de Software, e define seis níveis de capacidade, sequenciais e cumulativas que podem ser utilizadas como uma métrica para avaliar como é que uma organização está a realizar um determinado processo e também podem ser utilizadas como um guia para a melhoria. Esta norma define também um guia para a orientação da melhoria de processo, tendo como referência um modelo de processo, e, como uma das etapas a realizar, a avaliação de processo. Este guia sugere 8 etapas sequenciais. Em seguida existem ciclos de melhoria, nos quais são identificados um conjunto de melhorias, uma avaliação das práticas correntes em relação à melhoria, um planeamento da melhoria, seguido da implementação, confirmação, manutenção e acompanhamento da melhoria.

Comparando um modelo com o outro, podemos dizer que o SPICE mede um processo específico através do nível de capacidade, enquanto o CMM mede a organização através do nível de maturidade [Paulk et al, 1995].

2.3.2.2.2 *Revisão dos Artigos Processos de Desenvolvimento*

➤ **Complexity Metrics for Measuring the Understandability and Maintainability of Business Process Models using Goal-Question-Metric (GQM) [Ghani, 2008]**

Objectivo: Este artigo fornece-nos um relatório completo sobre a forma como métricas de software existentes foram adaptadas, a fim de analisar a complexidade dos modelos de processo de negócio actuais. Foi também proposta uma *framework Goal-Question-Metric* (GQM) para medir a compreensibilidade e a sustentabilidade dos *Business Process Models* (BPMs).

Resumo: Os BPMs servem como uma base para a comunicação entre os *stakeholders* no processo de desenvolvimento de software. Para cumprir este objectivo, os BPMs devem ser fáceis de compreender e fáceis de manter. Por esta razão, é útil termos medidas que nos possam fornecer informações adequadas sobre a compreensibilidade e a sustentabilidade do BPM. Embora existam centenas de medidas de complexidade de software que têm sido descritas e publicadas por vários investigadores ao longo das últimas décadas, medir a complexidade dos modelos do processo de negócio é uma área de investigação bastante nova, com apenas um pequeno número de contribuições [Ghani, 2008]. O autor analisou e resumiu a adaptação de métricas complexas de Engenharia de Software que foram aplicadas em modelos de processos de negócio em 5 categorias: tamanho, complexidade, estrutura, compreensibilidade e modularização (ver tabela 2.12).

Métricas complexas de software	Métricas complexas de BPM	Uso da métrica em BPM
Lines of code	Number of activities	Conta simplesmente o número de actividades do modelo
Cyclomatic number [McCabe, 1976]	<i>Control Flow Complexity</i> (CFC)	Mede o número de fluxos de controlo no modelo
Nesting depth [Volker, 2007]	<i>Nesting depth</i>	Define a estrutura do modelo; quanto maior o valor do <i>Nesting depth</i> mais complexo é o modelo.
Knot-count [Mendling, 2007]	<i>Split-join ratio</i>	Define se o modelo está bem estruturado
Cognitive Weight [Cardoso, 2006]	<i>Cognitive Weight</i>	Mede a compreensibilidade de um modelo
Anti-pattern [Volker, 2007]	<i>Anti-pattern</i>	Permite descobrir o mau estilo de modelação no modelo BPM
Fan-in/Fan out [Henry and Kafura, 1981]	<i>Fan-in/Fan out</i>	Permite definir a boa ou má modularização de um modelo

Tabela 2.12: Métricas complexas para software e BPMs

Neste artigo também foi feita uma comparação entre as abordagens *Goal-Attribute-Measure* (GAM), *Balanced Scorecard* (BSC) e *Goal-Question-Metric* (GQM). Concluiu-se que o GAM é melhor utilizado para medir objectivos específicos; o BSC é utilizado para medir o progresso organizacional, mesmo que fora do âmbito da TI e o GQM é genérico, embora historicamente usado em Engenharia de Software, é usado para identificar, a partir dos objectivos, quais as métricas que os podem ajudar a avaliar (ver tabela 2.13).

Abordagem	Arquitectura	Âmbito	Foco
GAM	Objectivo Atributo Medida	Medição do objecto	Atributo Estruturação Definição
BSC	Objectivo <i>Driver</i> Indicador	Organização	Definição de <i>Driver</i>
GQM	Objectivo Questão Métrica	Projecto	Definição da pergunta

Tabela 2.13: Comparação entre as abordagens GAM, BSC e GQM

Crítica: Este artigo expressa métricas complexas de Engenharia de Software que podem ser definidas para o modelo de processo de negócio. A abordagem GQM garante que cada métrica tem uma finalidade, e que nenhuma métrica é definida sem um propósito. O artigo propõe métricas com o intuito de avaliar a compreensibilidade¹⁰ e a facilidade de manutenção de modelos BPM. Contudo, a

¹⁰ **Compreensibilidade** - grau para o qual o propósito do sistema ou do componente é claro para o avaliador [Boehm, 1978]

Facilidade de manutenção – é a facilidade com que um sistema ou componente de software pode ser modificado para corrigir falhas, melhorar o desempenho ou outros atributos, ou se adaptar a um ambiente alterado, *IEEE Standard Computer Dictionary*.

abordagem GQM não permite gerar um conjunto mínimo de métricas. Além disso, a maioria das métricas adaptadas sobre BPM não foram validadas.

Classificação:

Critérios	Ghani, 2008
Domínio da Aplicação	GSS
Aplicabilidade / Replicação	CP
Diagnóstico	CL
Formalização/ Discriminação	CP
Validação	NC

Tabela 2.14: Classificação baseada nos critérios definidos na taxionomia

➤ CMM Versus SPICE Architectures [Paulk et al., 1995]

Objectivo: Desenvolvido pelo SEI (*Software Engineering Institute*) o CMM para software tem popularizado o conceito de medir a maturidade do processo de software das organizações. Um dos objectivos do SPICE é criar uma forma de medição da capacidade do processo, embora não utilizando uma abordagem específica, como os níveis de maturidade do SEI. A abordagem seleccionada pretende medir a implementação de processos específicos. Utilizando esta abordagem, os níveis de maturidade podem ser vistos como conjuntos de perfis de processos. Este artigo ainda discute os prós e contras desta abordagem.

Resumo: O CMM descreve os princípios e as práticas subjacentes à maturidade dos processos de software e destina-se a ajudar organizações de software a melhorarem a maturidade dos seus processos de software em termos de um caminho evolucionário, desde processos *ad hoc* e caóticos, até processos maduros e disciplinados. O CMM está organizado em cinco níveis maturidade: 1. Inicial; 2. Repetível; 3. Definido; 4. Gerido; 5. Optimizado. Excepto para o nível 1, cada nível de maturidade é decomposto em várias áreas de processo chave que indicam as áreas que uma organização deve focar para melhorar o seu processo de software. Cada área de processo chave é descrita em termos de práticas-chave, que descrevem a infraestrutura específica e as actividades que mais contribuem para a efectiva implementação da área de um processo chave. Estas práticas-chave são organizadas por características comuns. As características comuns são atributos que indicam se a implementação da área de um processo chave é efectiva, repetível e duradoura.

O SPICE pode ajudar as organizações na capacidade, ajudando uma organização, a (i) determinar a capacidade de um potencial fornecedor de software, (ii) a melhorar o seu próprio desenvolvimento de software e processos de manutenção, e (iii) na auto-avaliação, a determinar a sua capacidade para implementar um novo projecto de software. O conjunto básico do SPICE que abrange a avaliação de processos de software, é constituído por 7 guias, entre os quais, o *Baseline Practices Guide* (BPG). Este guia define, a um nível elevado, os objectivos e as actividades fundamentais que são essenciais para uma boa Engenharia de Software. O BPG do SPICE é o equivalente ao CMM do SEI. O BPG descreve que actividades são exigidas, e não como elas devem ser implementadas. No BPG existem

seis níveis de capacidade: (0) Nível Não-Desempenhado; (1) Nível Informalmente-Desempenhado; (2) Nível Planeado-e-Definido; (3) Nível Bem-Definido; (4) Nível Quantitativamente-Controlado; e (5) Nível Melhoria-Contínua. No BPG, a capacidade dos processos evolutivos é expressa em termos de níveis de capacidade, características comuns, e práticas genéricas. Um nível de capacidade é um conjunto de características comuns (conjuntos de actividades) que trabalham em conjunto para proporcionar uma grande melhoria na capacidade para desempenhar um processo. Uma característica comum é um conjunto de práticas que abordam o mesmo aspecto do processo de implementação. As características comuns do BPG são agrupadas de acordo com os níveis de capacidade. Uma prática genérica é uma prática de implementação que aumenta a capacidade de desempenhar qualquer processo.

Crítica: Tanto o CMM, que descreve a maturidade dos processos de software, como o BPG, que descreve a evolução de um processo, têm vantagens e desvantagens.

Os níveis de capacidade do BPG e os níveis de maturidade do CMM são semelhantes, mas distintos. Os níveis de capacidade são aplicados tendo um processo por base. Os níveis de maturidade organizacionais podem ser definidos como um conjunto de perfis para esses processos. Uma das vantagens da proposta BPG é que mede a evolução de cada processo separadamente dos outros processos.

Em contrapartida, as áreas do processo chave do CMM são definidas como residentes num único nível de maturidade. Contudo, os processos maduros são descritos em diferentes áreas do processo chave e em níveis diferenciados. Uma das vantagens da arquitectura do CMM é que ela incide sobre as "poucas áreas vitais" que normalmente bloqueiam o desempenho de um processo numa fase particular na vida da organização. Uma desvantagem da arquitectura do BPG é que as questões menos importantes sobre processos podem sair do grupo das "questões vitais" à medida que existem confrontos nas prioridades de melhorias. A desvantagem da arquitectura CMM é que as pessoas podem perder de vista os processos que não são cruciais de um determinado nível de maturidade, mas que ainda assim devem ser executados.

A solução mais simples para integrar estes dois modelos seria construir um modelo faseado em cima de uma arquitectura contínua. Os níveis de maturidade do CMM podem ser camadas em cima dos níveis de capacidade do BPG, como está ilustrado, num nível elevado de abstracção, na figura 2.5.

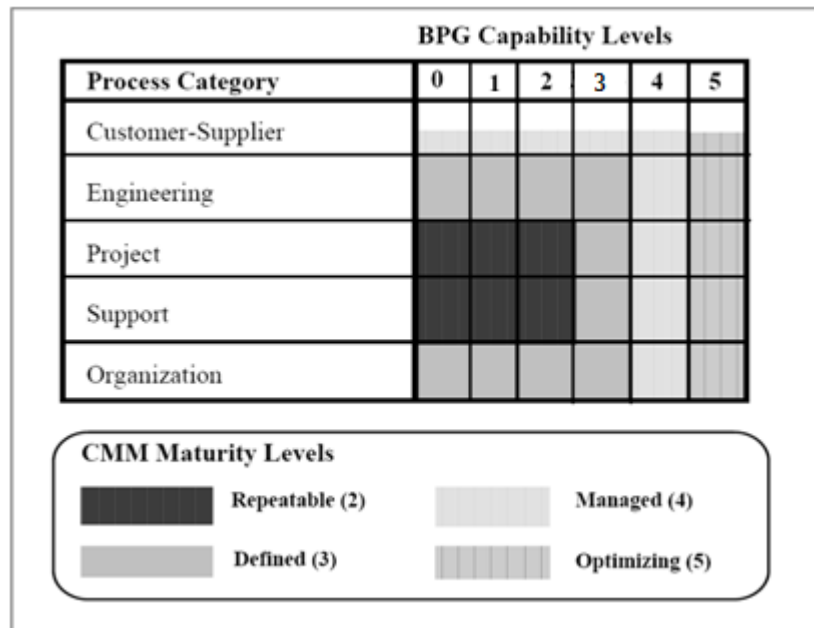


Figura 2.5: Integração dos dois modelos (extraído de [Paulk et al., 1995])

Outra desvantagem do BPG é que pode ser difícil decidir quais as questões que devem ser “atacadas” em primeiro lugar. Uma desvantagem do CMM é que prescreve, atacando questões de gestão de projectos antes dos de engenharia (apesar desta priorização ser baseada na experiência de processos de melhoria bem sucedidos). Poderíamos então dizer, que o BPG descreve o terreno, enquanto que o CMM fornece um *roadmap*.

Muitos dos conceitos expressos no nível superior das áreas de processo chave do CMM, nomeadamente no nível 4, são representados como características comuns e práticas genéricas no BPG. Uma desvantagem do BPG é o potencial grande número de decisões na avaliação e classificação de um processo. Há 26 práticas genéricas e 35 processos, o que sugere potencialmente 910 decisões de avaliação, mesmo que se ignorem 5 ou 6 práticas base por processo. O CMM contém 316 práticas chave. Portanto, mesmo que se avaliem “apenas” as 112 actividades realizadas nos níveis 2 e 3, isso pode ser uma tarefa que pode envolver um esforço considerável. Desenvolver um método de avaliação que possa efectivamente usar o BPG na avaliação e classificação do conjunto completo de processos do BPG é um dos desafios que ainda enfrenta o projecto SPICE [Paulk e tal., 1995].

Classificação:

Critérios **Paulk, Konrad and Garcia, 1995**

Domínio da Aplicação	GSS
Aplicabilidade / Replicação	CP
Diagnóstico	NC
Formalização / Discriminação	CL
Validação	NC

Tabela 2.15: Classificação baseada nos critérios definidos na taxionomia

2.4 Resumo da Taxionomia e do Estado da Arte

É preciso ter presente que a decisão da escolha dos artigos pode limitar posteriormente, quer a visualização dos resultados, quer a sua análise, pelo que é importante a identificação clara do contexto para que cada documento foi idealizado.

Podemos concluir, a partir da análise da tabela 2.16, que apenas encontramos um artigo que interliga a governança de TI com a maturidade dos processos (número 7 - sombreado com quadrados azuis na tabela 2.16). Por esta razão, vamos adoptar este modelo para nos ajudar na orientação da execução do nosso objectivo nesta dissertação.

Em todos os documentos analisados sobre a governança de TI (do 1 ao 6 - sombreado com linhas horizontais verdes na tabela 2.16), foi possível encontrar abordagens inovadoras e importantes para o desenvolvimento de uma ferramenta, que nos ajude a definir arquétipos de governança de TI. Constatamos que são poucos os artigos que nos fornecem técnicas de levantamento e avaliação do estado actual da empresa. O artigo de *Hoffman and Weill* (número 3 na tabela 2.16), de 2007 é dos poucos artigos que tem uma descrição completa sobre a forma como devemos fazer um diagnóstico a uma organização. Por esta razão, e também pelo facto de cobrir largamente a validação do modelo apresentado através de evidências e trabalho experimental, e nos mostrar que é possível aplicar a teoria à prática, escolhemos este artigo para nos guiar na elaboração de questionários e na definição de arquétipos de governança.

Pelas propostas apresentadas sobre a maturidade dos processos (do 8 ao 14 - sombreado com linhas verticais vermelhas na tabela 2.16) observamos que, recentemente, se têm vindo a procurar e a usar técnicas tanto de modelos de qualidade de serviços de TI como de modelos de qualidade de sistemas e software. Apesar disso, não existe nenhum documento que descreva um método que nos ajude a conhecer ou a determinar o grau de complexidade dos processos de uma empresa. Nestes artigos apenas são formalizados e discriminados os modelos de maturidade de processos em si e não a técnica para os alcançar. Também, são poucos os artigos que corroboram, com casos de estudo ou trabalho experimental, a validação destes modelos. Este é um dos motivos pelo qual escolhemos o artigo do *Debreceeny*, de 2006 (número 12 na tabela 2.16). Outro motivo é o facto de este artigo falar de modelos de qualidade, tanto de serviços de TI, como de sistemas e software.

Também seleccionamos o artigo de *Ghani* de 2008 (número 11), para nos ajudar a definir e formalizar um conjunto de métricas que permitam exprimir a complexidade de um processo.

	Crítérios Propostas	Domínio da Aplicação	Aplicabilidade e / Replicação	Diagnóstico	Formalização /Discriminação	Validaçã o
1	ISO 38500 2008	GTI	CP	NC	CP	NC
2	Jan Van Bon 2008	GTI	CL	NC	CL	CP
3	Hoffman and Weill 2007	GTI	CC	CC	NC	CL
4	Weill and Ross 2004	GTI	CL	CP	CP	CP
5	Weill 2004	GTI	CP	NC	CC	CP
6	Weill and Woodham 2002	GTI	CL	NC	CP	CP
7	Simonsson, Johnson 2008	GTI e GSTI	CL	CL	CP	CL
8	Woodhouse 2008	GSTI	CP	NC	CP	NC
9	Spremic, Zmirak, Kraljevic 2008	GSTI	CL	NC	CP	CL
10	Nicho, Cusack 2007	GSTI	CP	NC	CL	NC
11	Ghani 2008	GSS	CP	CL	CP	NC
12	Debreceeny 2006	GSS e GSTI	CP	NC	CL	CL
13	Paulk, Konrad, Garcia 1995	GSS	CP	NC	CL	NC

Tabela 2.16: Resumo do estado da arte

[Esta página foi intencionalmente deixada em branco]

Capítulo 3

Governança de TI

Conteúdo

3.1 INTRODUÇÃO	40
3.2 A IMPORTÂNCIA DA GOVERNANÇA DE TI.....	41
3.3 COMO AS EMPRESAS GOVERNAM A TI: DECISÕES-CHAVE E ARQUÉTIPOS DE TI.....	43
3.4 COMO É QUE AS DECISÕES SÃO FORMADAS E DIVULGADAS	47
3.5 PERFIS DE GOVERNANÇA DE TI	49
3.6 ABORDAGENS DA GOVERNANÇA DE TI.....	50
3.7 PLANEJAR E AVALIAR A GOVERNANÇA DE TI.....	54
3.8 RECOMENDAÇÕES PARA A GOVERNANÇA DE TI.....	55

Neste capítulo são apresentados alguns conceitos básicos sobre a governança de TI e qual a sua importância. Fala-se das principais decisões e arquétipos de TI, dos mecanismos de TI, dos perfis e abordagens da governança de TI e da forma como as empresas governam a TI. Finalmente, mostra-se como planejar e avaliar a governança de TI e apresentam-se algumas recomendações para a governança de TI.

3 Governança de TI

3.1 Introdução

O termo “governança corporativa” foi criado no início da década de 1990 nos Estados Unidos e na Grã-Bretanha, para definir as regras que regem o relacionamento, dentro de uma empresa, dos interesses de accionistas maioritários e minoritários e administradores [Charan, 2005].

A governança corporativa pode ser definida como: *“a criação de uma estrutura que determina os objectivos organizacionais, e onde deve ser feita a monitorização do desempenho para assegurar a concretização dos objectivos da empresa”* [OECD, 1999].

Isto significa que as empresas que têm governança corporativa possuem uma estrutura de processos que assegurará que os objectivos organizacionais sejam alcançados, principalmente pela monitorização do desempenho da empresa sobre o mercado em que actua.

Contudo, não existe um modelo único de uma boa governança corporativa. Deve existir uma equipa de gestão de topo que trabalhe para implementar os princípios de governança, que assegure a eficiência dos processos organizacionais [Weill, 2006].

Um elemento fundamental da governança corporativa é a “governança de TI”, responsável pela gestão de tecnologia da empresa no que diz respeito a investimentos, projectos, gerar indicadores de desempenho e principalmente suportar os processos de negócios da empresa [Pereira, 2007].

Uma definição proposta em [Bon, 2008] sobre governança de TI é: *“a obrigação de prestar contas, de ter responsabilidade e de conceber a organização de TI, tendo como objectivo uma utilização eficaz e eficiente da TI dentro de processos de negócio, e que obedeça às normas internas e externas”*.

A gestão sobre o emprego da tecnologia para suportar processos de negócios representa um dos mais valiosos recursos das empresas. Além disso, num ambiente de negócios altamente competitivo e dinâmico é requerida uma habilidade de gerir TI para suportar as tomadas de decisões de forma rápida, constante e com custos cada vez menores.

Desta forma, somente com a gestão de recursos tecnológicos é que se pode ter uma governança corporativa [Pereira, 2007].

A governança corporativa necessita de uma visão geral da corporação e é através desta visão que os executivos têm condições para dar as devidas direcções e traçar os objectivos para actuação da organização no mercado [Botto, 2004].

A governança em TI suportará a governança corporativa, se estiver ciente dos objectivos corporativos traçados e fizer a gestão dos recursos tecnológicos necessários para suportar todo o negócio da empresa.

Em conclusão, a governança de TI tem por objectivos básicos conhecer os valores e a importância estratégica da TI, assegurar que pode suportar as operações da empresa e garantir que pode implementar as estratégias necessárias para atender o crescimento e a expansão da empresa no futuro.

As melhores práticas de governança de TI visam garantir que as expectativas da TI sejam atendidas e os riscos inerentes a TI sejam minimizados.

3.2 A Importância da Governança de TI

A governança de TI é importante porque influencia os benefícios recebidos dos investimentos de TI. Através de uma combinação de práticas (tais como redesenhar os processos de negócio e delinear bem os mecanismos de governança) e de investimentos de TI adequadamente articulados, as empresas de alto desempenho geram retornos superiores sobre os seus investimentos de TI [Weill, 2004]. Uma estimativa é de um retorno até 40% maior do que os seus concorrentes para o mesmo investimento de TI [Brynjolfsson, 2002], [Weill and Broadbent, 1998].

Estas empresas de alto desempenho procuram pro-activamente o valor da TI de, pelo menos, cinco maneiras diferentes: (1) definindo as estratégias de negócio e o papel que a TI desempenha na consecução dos mesmos; (2) medindo e gerindo o valor gasto e o valor recebido da TI; (3) desenhando práticas organizacionais para adaptar a TI às suas estratégias de negócio; (4) atribuindo a responsabilidade pelas mudanças organizacionais necessárias para beneficiar de novas capacidades de TI; (5) aprendendo a partir de cada implementação e partilhando e reutilizando os activos de TI [Weill, 2004].

Em parte, as empresas de alto desempenho têm êxito onde outras falham, ou seja, na implementação da governança de TI eficaz para apoiar as suas estratégias e institucionalizar boas práticas [Weill and Broadbent, 2003].

Para além da correlação entre o desempenho das organizações e a governança de TI, e segundo o estudo feito por Peter Weill no *MIT Sloan School of Management* em 2004 [Weill, 2004], verificou-se que as empresas com desempenho de governança de TI acima da média, que seguem uma estratégia específica (como a proximidade com o cliente), tinham um lucro 20% maior do que as empresas com um governança pobre seguindo a mesma estratégia.

Existe uma série de definições interessantes e fontes de informação sobre a governança TI. Por exemplo, o *IT Governance Institute* (www.itgi.org) define governança de TI como "*uma estrutura de relações e processos para controlar a empresa, a fim de alcançar os objectivos da empresa, acrescentando valor ao equilibrar os riscos versus o retorno sobre a TI e os seus processos*" [IT Governance Institute, 2000].

Outra definição e perspectiva é a de Wim Van Grembergen: "*A governança de TI é a capacidade organizacional exercida pelo Conselho de Administração, pela gestão de executivos e pela gestão de TI para controlar a formulação e implementação da estratégia da TI e, desta forma, assegurar a fusão do negócio e da TI.*" [Wim Van Grembergen, 2007].

Contudo, estas definições são geralmente consistentes no propósito com a definição de Peter Weill, mas diferem na forma. Peter Weill define governança de TI como "*especificar a framework de direitos*

de decisão e a obrigação de prestar contas para encorajar comportamentos desejáveis na utilização da TI”.

Um comportamento desejável neste contexto é o que for consistente com a missão, a estratégia, os valores e as normas e cultura da organização, tais como o comportamento na promoção do espírito empresarial, a partilha e a reutilização, ou a redução de custos implacável.

A governança determina sistematicamente quem faz cada tipo de decisão (direito de decisão¹¹), quem contribui para uma decisão (direito de contribuição¹²) e como essas pessoas (ou grupos) são responsabilizados pelo seu papel. Como já foi referido, a boa governança de TI inspira-se nos princípios de governança corporativa para gerir e usar a TI para alcançar objectivos de desempenho corporativos. A governança eficaz de TI estimula e impulsiona todos os colaboradores das empresas, na habilidade no uso da TI, garantindo simultaneamente o cumprimento da visão global e dos princípios da organização. Como resultado, a boa governança de TI torna-se, simultaneamente, poderosa e controladora.

Todas as organizações têm governança de TI. A diferença é que as empresas com uma governança eficaz têm activamente delineado um conjunto de mecanismos de governança de TI (por exemplo, comités, processos de orçamentação, aprovações, estrutura organizacional de TI, recuperação de custos de TI, etc.) que incentivam comportamentos coerentes com a missão, estratégia, valores, normas e cultura da organização. Nestas empresas, quando o "comportamento desejável" muda, a governança de TI também muda [Weill, 2004].

A governança de TI não pode ser considerada isoladamente, pois tem ligações com a governança de outros activos chave da organização (tais como financeiros, humanos, propriedade intelectual, etc.). A governança dos activos chave, por sua vez, tem ligações com a governança da organização e com os comportamentos desejáveis [Weill and Woodham, 2002].

Em parte, o estímulo no estudo da governança de TI apareceu, pois acreditamos que a governança financeira na maioria das empresas poderia fornecer um bom modelo de governança de TI. Os Directores Financeiros (CFO - *Chief Financial Officer*), não tomam todas as decisões financeiras ou assinam todos os cheques. Em vez disso, planeiam a governança para identificar quem pode tomar as decisões financeiras e como essas pessoas são responsabilizadas. Além disso, os Directores Financeiros têm ferramentas para gerir o seu portfolio de investimento, perfil de risco, *cash flow*, tendo em atenção os indicadores de desempenho. A mesma abordagem pode ser aplicada à governança de TI.

¹¹ Os direitos de decisão são detidos por grupos diferentes para cada decisão.

¹² Os direitos de contribuir para uma decisão envolvem muitas pessoas.

3.3 Como as Empresas Governam a TI: Decisões-chave e Arquétipos de TI

Os gestores tomam centenas de decisões numa semana – alguns, após uma análise cuidadosa e outros, como parte da sua actividade diária frenética. O planeamento e análise da governança exige um processo de revisão, a partir de tomadas de decisão do dia-a-dia, centrando-se na identificação de tomadas de decisão fundamentais e quem deve fazê-las. Peter Weill e os seus colaboradores propuseram que as grandes empresas têm cinco domínios de decisão de TI para fazer (ver tabela 3.17) e devem utilizar seis arquétipos¹³ de governança mutuamente exclusivos (tabela 3.19) para tomar essas decisões.

Domínios de decisão	Definição	Referências às citações
Princípios de TI	São declarações de alto nível sobre como as TI's serão usadas para criar valor de negócio.	[Davenport and Metsisto, 1989] [Broadbent and Weill, 1997]
Arquitectura de TI e Standards	Um conjunto integrado de opções técnicas para orientar a organização na satisfação das necessidades do negócio. A arquitectura é um conjunto de políticas e normas para o uso da TI e delineia um caminho para a forma como o negócio será feito (inclui dados, tecnologia e aplicações)	[Keen, 1995] [Ross, 2003]
Estratégia de Infra-estrutura de TI	Estratégias para a orçamentação da capacidade de TI (tanto técnica como humana), partilhada em toda a empresa como serviços fiáveis e coordenados centralmente (por exemplo, rede, <i>help desk</i> e dados partilhados)	[Keen, 1991] [Weill, Subramani and Broadbent, 2002]
Necessidade de aplicações de negócio	Especifica a necessidade do negócio em adquirir ou desenvolver internamente as aplicações de TI	[Earl, 1993]
Investimento de TI	Decisões sobre quanto e onde investir em TI, incluindo a sua justificação, aprovação e responsabilização.	[Devaraj and Kohli, 2002] [Ross and Beath, 2002]

Tabela 3.17: 5 Domínios de Decisão de TI (adaptado de [Weill, 2004])

Cada uma dessas decisões de TI pode ser tomada por gestores corporativos, gestores de unidades de negócio ou gestores funcionais (ou alguma combinação destes). Assim, o primeiro passo na concepção da governança de TI é o de determinar quem deve tomá-las, e ser responsabilizado por cada domínio de decisão. Para ajudar a pensar sobre quem deve tomar essas decisões, a tabela 3.18, apresenta uma amostra de perguntas que cada domínio de decisão deverá abranger [Weill and Ross, 2004].

¹³ Os arquétipos descrevem as combinações de pessoas que têm direitos de decisão ou que contribuem para a tomada de decisões de TI, ou seja, os arquétipos definem como é que os grupos gestores podem ser organizados para o processo de tomada de decisão.

Princípios de TI	<p>Como é que os princípios de negócio se traduzem em princípios de TI para orientar a tomada de decisões de TI?</p> <p>Qual é o papel da TI no negócio?</p> <p>Quais são os comportamentos desejáveis da TI?</p> <p>Como é que a TI será financiada?</p>
Arquitectura de TI e Normas	<p>Quais são os principais processos de negócio da empresa? Como é que eles se relacionam?</p> <p>Qual é a informação que impulsiona esses processos fundamentais? Como é que estes dados devem ser integrados?</p> <p>Que capacidades técnicas devem ser normalizadas a nível da empresa para suportar a eficiência da TI e facilitar o processo de normalização e de integração?</p> <p>Que actividades devem ser normalizadas a nível da empresa para suportar a integração de dados?</p> <p>Que opções tecnológicas devem orientar a abordagem da empresa para iniciativas de TI?</p>
Estratégia de Infraestrutura de TI	<p>Quais são os serviços de infraestrutura mais críticos para alcançar os objectivos estratégicos da empresa?</p> <p>Que serviços de infraestrutura devem ser implementados a nível da empresa e quais são os seus respectivos requisitos de nível de serviço?</p> <p>Como é que os serviços de infraestrutura devem ser taxados (preço)?</p> <p>Qual é o plano para manter as tecnologias subjacentes actualizadas?</p> <p>Que serviços de infraestrutura devem ser subcontratados externamente?</p>
Necessidade de aplicações de negócio	<p>Quais são as oportunidades de processos de negócio e do mercado para novas aplicações de negócio?</p> <p>Como é que as experiências estratégicas são planeadas para avaliar o sucesso?</p> <p>Quando é que uma necessidade de negócio justifica uma excepção ao normativo?</p> <p>Quem será o proprietário dos resultados de cada projecto e quem irá instituir as alterações organizacionais para garantir o seu valor?</p>
Investimento de TI	<p>Quais são as alterações ou melhorias de processos que são estrategicamente mais importantes para a empresa?</p> <p>Qual é a distribuição no portfolio actual da TI? Este portfolio é consistente com os objectivos estratégicos da empresa?</p> <p>Qual é a importância relativa do nível das empresas versus investimentos da unidade de negócio? As práticas de investimento actual reflectem a sua importância relativa?</p>

Tabela 3.18: Questões chave para cada decisão de TI (adaptado de [Weill and Ross, 2004])

Cada um desses domínios de decisão de TI tem sido objecto de numerosos artigos e livros¹⁴, Estes domínios estão altamente inter-relacionados, mas uma empresa tem frequentemente arquétipos de governança diferentes para os diferentes domínios [Weill and Woodham, 2002].

Com base nos trabalhos sobre governança corporativa, estado da governança e políticas de informação, usamos arquétipos políticos (monarquia, feudal, federal, duopólio e anarquia) para descrever as combinações de pessoas que têm, quer direitos de decisão, quer direitos de contribuir para a decisão de TI [Davenport, 1997 – capítulo 5]. Cada um dos nossos seis arquétipos (descritos na

¹⁴ Para uma visão geral, ver: [McKeen and Smith, 2003] e [Zmud, 2000].

tabela 3.19) pode conter direitos de decisão ou direitos de contribuir para a decisão de cada um dos cinco principais domínios de decisão de TI. A governança pode (e deve) ser concebida e avaliada em vários níveis numa grande empresa, tais como *enterprise-wide*, unidade de negócio e, talvez, por região ou grupo de unidades de negócio. Este trabalho centra-se na análise do perfil de governança de TI, mas a mesma abordagem pode ser usada noutros níveis organizacionais.

Arquétipos	Os direitos de decisão ou direitos de contribuir para uma determinada decisão de TI são detidos por:	Directores ¹⁵	Chefias ¹⁶	Coordenadores ¹⁷
Monarquia do negócio	Executivos de negócio individuais ou grupos (directores). Inclui Comité de Executivos e Conselhos de TI com os executivos de negócio seniores. Exclui executivos de TI que agem independentemente.	☒		
Monarquia de TI	Executivos de TI individuais ou grupos.		☒	☒
Feudal	Líderes de unidade de negócio, donos de processos-chave ou seus delegados			☒
Federal	Executivos de negócio e pelo menos um outro grupo de negócio (ex: directores e líderes de unidade de negócio). Os Executivos de TI podem participar.	☒	☒	☒
Duopólio de TI	Executivos de TI e um outro grupo (ex: directores ou líderes de unidade de negócio).	☒	☒	☒
Anarquia	Cada utilizador individual			

Tabela 3.19: Arquétipos de governança de TI (adaptado de [Weill, 2003])

Monarquia do negócio

Na monarquia do negócio são os executivos de negócio que tomam as decisões de TI que afectam toda a empresa [Weill, 2004].

Tipicamente, as monarquias do negócio recebem contributos para as decisões-chave de muitas fontes. Por exemplo, em muitas empresas, os contributos para decisões de investimento de TI são provenientes: (i) directamente do CIO; (ii) das unidades de negócios, através da equipe de liderança da TI; (iii) do processo de gestão de orçamento de TI; (iv) de acordos de níveis de serviço e da recuperação de custos de TI; (v) de um sistema de acompanhamento que mostra todos os recursos de TI e como eles são utilizados.

¹⁵ **Directores:** Director Geral (CEO – *Chief Executive Officer*), Director de Operações (COO – *Chief Operating Officer*), Director Financeiro (CFO – *Chief Financial Officer*), Director dos Recursos Humanos (CHRO – *Chief Human Resources Officer*), Director de Informática (CIO – *Chief Information Officer*), etc.

¹⁶ **Chefias:** Dependendo da estrutura organizacional temos líderes de unidades de negócio ou líderes dos vários departamentos que constituem a organização.

¹⁷ **Coordenadores:** Donos de processos de negócio - pessoa que possui responsabilidade e autoridade pela operação e melhoria do processo como um todo.

Monarquia de TI

Numa monarquia de TI, os profissionais de TI tomam as decisões de TI [Weill, 2004]. Por exemplo, o comitê de governança de TI, constituído por gestores de TI seniores, toma as decisões estratégicas que afectam a arquitectura TI.

As empresas implementam monarquias de TI de muitas formas diferentes, muitas vezes envolvendo profissionais de TI, tanto de equipas corporativas, como de unidades de negócio. Por exemplo, considere-se uma empresa que tem um grupo de arquitectura de TI com representantes de várias regiões, de todas as unidades estratégicas de negócio e de todos os centros de competência. Este grupo propõe "regras" de arquitectura à equipa de gestão sénior de TI, constituída pelo CIO corporativo e pelos CIOs das maiores unidades de negócio. Esta equipa garante que as regras fazem sentido para o negócio e responsabiliza-se por fazer cumprir as normas arquitectónicas.

Feudal

O modelo feudal é baseado na tradição "*merrie olde England*", onde cada príncipe e princesa, ou os seus cavaleiros designados, tomam as suas próprias decisões para otimizar as suas necessidades locais [Weill, 2004]. Para a governança de TI, o estado feudal é, tipicamente, uma unidade de negócio, uma região ou uma função. O modelo feudal, não sobressai, neste estudo, porque a maioria das empresas estavam à procura de sinergias entre unidades de negócio.

Federal

O modelo de tomada de decisão federal tem uma longa tradição no governo. O regime federal tende a equilibrar as responsabilidades e a obrigação de prestar contas de múltiplos órgãos do governo abrangendo, pelo menos, dois níveis hierárquicos, como o país e os estados. Charles Handy, e outros escritores de gestão, têm identificado a utilidade do modelo federal na negociação dos interesses, tanto de toda a empresa, como das unidades de negócio individuais [Handy, 1992].

Peter Weill define o modelo federal como aquele que coordena a tomada de decisões envolvendo tanto o centro como as suas unidades de negócio (pelo menos, dois níveis da hierarquia de negócio). As unidades de negócio representantes num modelo federal podem ser os líderes de unidade de negócio, ou os donos de processos de negócio, ou ambos. Os líderes de TI, tanto de unidades de negócio como corporativos, também podem participar. Nestes casos, são acrescentados ao grupo federal, mas não tomam o lugar de um dos grupos de negócio.

O modelo federal é, provavelmente, o arquétipo mais difícil de usar para a tomada de decisão, pois os líderes das empresas têm diferentes perspectivas dos líderes das unidades de negócio. Os requisitos dos níveis das empresas podem, e isso ocorre muitas vezes, colidir com os requisitos das unidades de negócio, exigindo compromissos para promover a concorrência. Além disso, os sistemas de incentivo focam, frequentemente, a atenção dos gestores nos resultados das unidades de negócio e não nos resultados da empresa.

O impacto de recursos partilhados no desempenho da unidade de negócio - e, especificamente, a transferência de preços praticados para esses recursos - normalmente suscita preocupações com a imparcialidade [Weill, 2004].

Duopólio de TI

O duopólio de TI é um regime de dois partidos, onde as decisões representam um acordo entre executivos de TI e um grupo de negócio¹⁸. Os executivos de TI podem ser qualquer combinação de grupos de TI. O grupo de negócio é constituído, tipicamente, por directores, líderes de unidades de negócio ou donos de processos de negócio.

Um duopólio de TI inclui sempre uma representação de TI (uma parte do duo) e uma representação de negócios (a segunda parte do duo) - mas apenas um grupo de representantes de negócios em cada duopólio (normalmente uma unidade de negócio ou um grupo de directores).

Aproximadamente 15% das respostas aos inquéritos utilizam duopólio de TI para tomar decisões nas três decisões de TI menos técnicas: princípios de TI, necessidade de aplicações de negócio e investimentos de TI. O arquétipo duopólio é popular, em parte porque envolve apenas duas partes de decisão. O duopólio pode atingir muitos dos objectivos de um modelo federal, usando uma estrutura de tomada de decisão simples de dois sentidos, em vez de uma estrutura multi-sentidos.

Do mesmo modo, um duopólio de TI tem a vantagem sobre o modelo feudal, pois o grupo central de TI é normalmente um dos poucos grupos que vê a empresa como um todo e pode olhar para as oportunidades de partilha e reutilização entre grupos de negócio e unidades de negócio.

Os duopólios de TI confiam, muitas vezes, na relação de gestores ou directores de unidade de negócio para representar as necessidades da unidade de negócio. O grupo corporativo de TI pode ter um duopólio com cada unidade de negócio, permitindo mais decisões adaptadas em menos tempo. Cada um destes duopólios tem também a vantagem de se concentrar directamente sobre as necessidades da sua unidade de negócio, resultando numa maior satisfação da unidade de negócio. Mas, ter um duopólio de TI com cada unidade de negócio pode ser caro e ineficaz, quando questões sobre o nível da organização estão a ser decididas [Weill, 2004].

Anarquia

Numa anarquia, os indivíduos ou grupos muito pequenos tomam as suas próprias decisões, apenas com base nas suas próprias necessidades. Eles diferem dos decisores do modelo feudal no tamanho da sua organização. Os feudais falam para grupos maiores; os anarquistas falam para pequenos grupos, muitas vezes só eles próprios. Anarquias são a desgraça de muitas organizações de TI, porque se envolvem descontroladamente e são caras para se suportarem e se tornarem seguras.

As anarquias não apareceram no nosso estudo. Mas elas existiram e foram apoiadas, por locais ou clientes individuais que exigiam uma resposta de TI muito rápida [Weill, 2004].

3.4 Como é que as decisões são formadas e divulgadas

Uma matriz completa de governança numa página mapeia os tipos de decisões, os arquétipos para a tomada dessas decisões e como essas decisões serão tomadas e monitorizadas. Esta última etapa requer a concepção e implementação de um conjunto coordenado de mecanismos de governança, tais

¹⁸ Duopólio, significa "controlo ou dominação por duas pessoas ou grupos", tal como definido em [Oxford University Press, 2003].

como comités, processos de orçamento, acordos de níveis de serviço (SLAs), recuperação de custos de TI, processos de arquitectura, etc. (ver tabela 3.20). É com estes mecanismos que os gestores trabalharam diariamente e a governança é promulgada [Weill and Ross, 2004].

Mecanismos de Governança	Constituição e Objectivos
Conselho de Administração	Constituído maioritariamente pelos directores dos vários sectores da organização. Tem uma visão holística.
Conselho de Gestão de TI	Órgão misto composto por executivos de TI e de negócio. Foca-se no valor do negócio.
Conselho de Administração de TI	Órgão exclusivamente composto por executivos de TI. Coordenação através da empresa.
Conselho da Arquitectura de TI	Identificação de tecnologias estratégicas
Relacionamento de gestores de negócio / gestores de TI	Garantia de <i>feedback</i> e boa interacção
Equipas de processos com membros de TI	Tem uma visão do processo
Acordos de Nível de Serviço (SLA) de TI	Especificação e quantificação dos serviços de TI
Acordos internos de recuperação de custos de TI	Forma de comportamento e recuperação de custos

Tabela 3.20: Mecanismos de governança de TI (adaptado de [Broadbent and Weill, 1998] [Weill and Woodham, 2002])

Os mecanismos de governança são os veículos utilizados para implementar diferentes arquétipos de governança. Eles podem ser específicos para um único domínio de TI ou podem abranger vários domínios [Broadbent and Weill, 1998] [Weill and Woodham, 2002].

Muitas vezes, o Conselho de Administração constitui um mecanismo onde as decisões mais importantes são tomadas. Isto encoraja uma visão holística mas, a menos que existam contributos de TI de alto nível, existe o risco de que as decisões possam não estar bem informadas. Os Conselhos de Gestão de TI reportam frequentemente ao Conselho de Administração e contêm membros partilhados entre estes dois órgãos. Tais conselhos podem proporcionar um ambiente centrado considerando vários níveis de políticas e de investimentos. Os Conselhos de Administração de TI são particularmente importantes para as grandes empresas, onde existe uma mistura de responsabilidades nos serviços de infraestrutura, nos níveis da empresa e outras nos níveis das unidades de negócio. O Conselho da Arquitectura de TI pode envolver a gestão de TI e do negócio na definição das orientações arquitectónicas. O Relacionamento de gestores de negócio / gestores de TI são prevalentes em muitas empresas, mas com uma variedade de nomes. Diariamente, eles desempenham um papel crítico na compreensão do funcionamento do negócio. Equipas de processos com membros de TI têm como objectivo ter uma visão do processo usando a TI efectivamente. O seu comportamento desejável é a gestão do processo. Os dois últimos mecanismos são caracterizados como um processo formal para assegurar que o comportamento diário é consistente com as políticas e fornece *input* para as decisões. Os Acordos de Nível de Serviço (SLA) de TI são parte do contracto de serviços entre duas ou mais entidades, no qual o nível da prestação de serviço é definido formalmente. Na prática, o termo é usado no contexto de tempo de entregas de um serviço ou de um desempenho específico. Os Acordos internos de recuperação de custos de TI têm como objectivo recuperarem os custos de TI do negócio.

O seu comportamento desejável é a responsabilidade do uso da TI [Broadbent and Weill, 1998] [Weill and Woodham, 2002].

Os mecanismos, transparentes, bem concebidos e bem compreendidos, promovem comportamentos de TI desejáveis e responsabilidade individual. Estes mecanismos implementam arranjos de governança pretendidos pelas empresas. Neste ponto, interessa questionar que modelo de governança irá funcionar melhor. Descobrimos que uma concepção de governança eficaz depende dos objectivos estratégicos de uma empresa e da sua estrutura organizacional. Assim, não podemos descrever o melhor modelo de governança. Porém, podemos aprender com os esforços das empresas de topo [Weill and Ross, 2004].

3.5 Perfis de Governança de TI

Cada uma das cinco principais decisões de TI tem uma escolha de seis arquétipos de governança, produzindo várias combinações possíveis. Os três perfis mais eficazes, medidos pelo desempenho da governança de TI, são apresentados na figura 3.6.

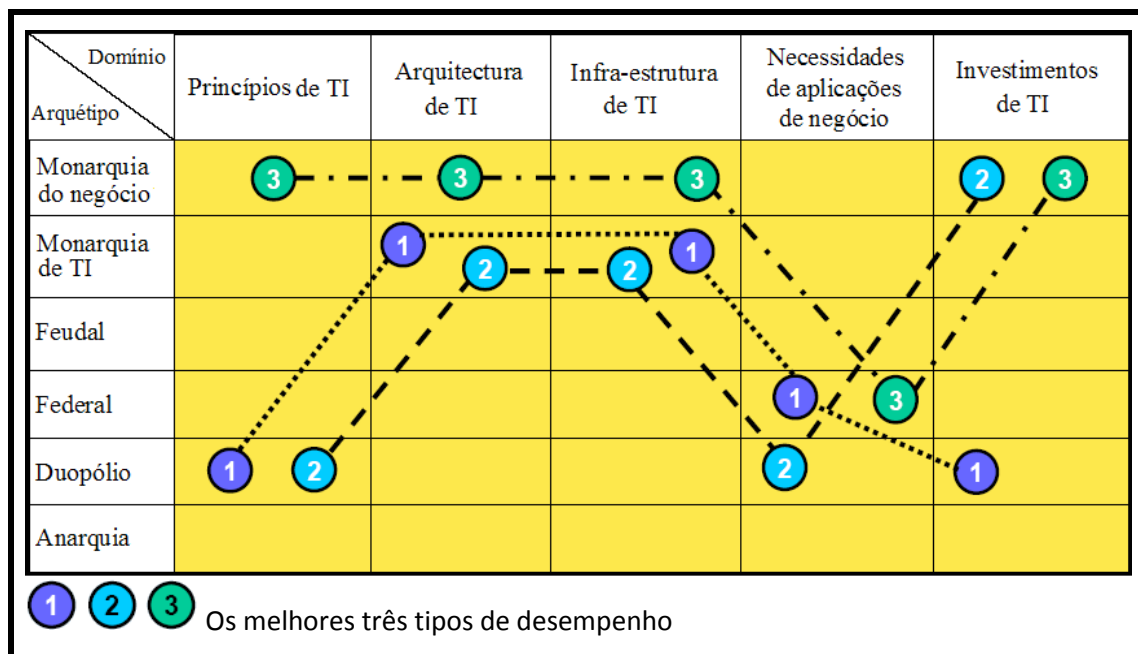


Figura 3.6: Os três perfis mais eficazes medidos pelo desempenho da governança de TI (adaptado de [Weill, 2004])

1. Perfil 1 - utiliza duopólio para os princípios e para os investimentos, monarquia de TI para a infraestrutura e arquitectura, e federal para as necessidades de aplicação do negócio. Este perfil exige grupos de TI que estão perfeitamente sintonizados com as necessidades do negócio, com um forte nível de confiança entre o negócios e a TI. O modelo federal, para as necessidades de aplicação, pode ser capitalizado em potenciais sinergias (tais como clientes comuns) nas unidades de negócio.
2. Perfil 2 - é semelhante ao perfil 1, utilizando um duopólio para as necessidades de aplicação e uma monarquia de negócio para o investimento. Para as empresas com poucas sinergias, usar um

duopólio para as necessidades de aplicação pode funcionar, porque há menos necessidade de coordenar todas as unidades de negócio.

Os perfis 1 e 2 são ambos bons pontos de partida, para as empresas equilibrarem o crescimento e a rentabilidade, porque as tensões das unidades de negócio procuram dar resposta às necessidades dos seus clientes locais. Estas necessidades estão bem equilibradas com os gestores seniores a governarem os investimentos de TI.

3. Perfil 3 - é muito mais centralizado, com as monarquias de negócio a assumirem todas as decisões, excepto as necessidades de aplicação do negócio (que é federal). Abordagens mais centralizadas são normalmente utilizadas em empresas com uma única unidade de negócio ou onde a rentabilidade ou o controlo do custo é um tema predominante. O perfil 3 exige líderes de negócio que estão interessados e bem informados sobre as questões de TI - muitas vezes o resultado da formação do CIO e de trabalharem em estreita colaboração com a equipa de gestão sénior. O perfil 3 também é sensível, quando estão a ocorrer grandes mudanças (por exemplo: fusões, grandes cortes de custos, crises, etc.) e os direitos de decisão devem ser bem defendidos.

A figura 3.6 ilustra a forma como, nos três perfis de alto desempenho, as cinco abordagens de tomada de decisão se encaixam, para criar um modelo de governança global, que está a reforçar e equilibrar as tensões inerentes às grandes empresas. Por exemplo, uma monarquia de TI para a arquitectura de TI pode ser muito eficaz, se a arquitectura for orientada pelos princípios de TI estabelecidos por uma monarquia do negócio ou por um duopólio. Na monarquia do negócio, os decisores da TI concentram-se na criação de uma arquitectura de TI integrada e flexível, orientada pelos princípios de TI do negócio, produzidos pelos líderes seniores [Weill, 2004].

3.6 Abordagens da governança de TI

A governança eficaz de TI deve ser visível nas métricas de desempenho do negócio. Embora muitos outros factores influenciem as medidas de desempenho financeiro, o forte desempenho dá confiança para que as empresas governem a TI eficazmente. Dadas diferentes estratégias e formas organizacionais, as variadas empresas tentarão incentivar comportamentos distintos [Weill and Ross, 2004].

Assim, os perfis de governança podem variar de abordagens mais centralizadas (notavelmente mais monarquias) para abordagens mais descentralizadas (notavelmente mais feudais), passando pelos modelos, federal e duopólio. Veja a tabela 3.21 para uma descrição destas diferentes abordagens. Da mesma forma, alguns mecanismos de governança apoiam abordagens mais centralizadas (como os Conselhos da Administração e o processo de aprovação de capital centralizado); outros apoiam mais abordagens híbridas (como o relacionamento de gestores de negócio / TI e os acordos de nível de

serviço (SLA))¹⁹; outros ainda apoiam abordagens descentralizadas, que implantam muito poucos mecanismos, estando focados, frequentemente, na gestão de riscos.

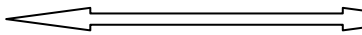
	Desempenho		
	Lucro	Utilização de activos	Crescimento
Orientação estratégica	Rentabilidade através da integração a nível da empresa. Foca-se nas competências essenciais.	Operação eficiente, incentivando a partilha e a reutilização	Incentivar a inovação das unidades de negócio com poucos processos exigidos por regra
Métricas chave	ROI ²⁰ /ROE ²¹ e Custos do processo de negócio	ROA ²² e custos da unidade de TI	Crescimento das receitas
Mecanismos de governança de TI chave	- Mecanismos de gestão a nível da empresa (por ex: Conselho da Administração) - Processo de arquitectura - Aprovação do capital - Acompanhamento do valor do negócio da TI	- Relacionamento de gestores de negócio / TI - Processo de equipas com os membros de TI - SLA e recuperação de custos de TI - Órgão de decisores da liderança de TI	- Aprovação do orçamento e gestão de risco - Responsabilidade local - Portais ou outras fontes de informação / serviços
Infra-estrutura de TI	Categorias de serviços partilhados centralmente	Serviços partilhados centralmente coordenados	Capacidades locais personalizadas com poucos requisitos de serviços partilhados
Princípios de TI chave	Baixa dos custos de negócio através de processos de negócios normalizados	- Baixa de custos da unidade de TI - Reutilização de modelos ou serviços normalizados	- Inovação local com práticas da comunidade - Serviços partilhados opcionais
Governança	Mais centralizado Ex: Monarquias e Federal		
		Ex: Federal e Duopólio	Mais descentralizado Ex: Regime Feudal e ênfase na gestão de risco

Tabela 3.21: Lições de governança para gestores de topo (adaptado de [Weill and Ross, 2004])

As empresas com alto desempenho governam de forma significativamente diferente da de outras empresas. Os gestores de topo variam na sua abordagem para a governança, relativamente a métricas de desempenho [Weill and Ross, 2004].

Características da governança com alto desempenho [Weill and Gartner, 2002]

- Estratégias mais focadas - maior diferenciação entre a proximidade com o cliente, a inovação dos produtos ou a excelência operacional;

¹⁹ Para uma discussão sobre modalidades de governança híbrida ver [Brown, 1997]

²⁰ ROI- *Return on Investment* = (Lucro Operacional/Investimentos) – medida que procura avaliar a atractividade económica do empreendimento, através da quantificação do resultado operacional produzido pela empresa, independentemente da forma como a mesma é financiada [Assaf Neto, 2003].

²¹ ROE – *Return on Equity* = (Lucro Líquido/Património Líquido) – o retorno sobre o Património Líquido é uma das principais medidas de rentabilidade, pois mede a decisão do accionista investir o seu capital na empresa, ao invés de aplicá-lo numa outra alternativa do mesmo risco [Iludícibus, 1998].

²² ROA – *Return on Assets* = (Lucro Operacional/Activo) – é uma medida que quantifica o resultado gerado pelas actividades operacionais da empresa e é calculado em relação ao investimento total, expresso pelo activo, o qual representa todos os bens e direitos da empresa oriundos tanto de passivos onerosos como de funcionamento, além dos recursos próprios [Iludícibus, 1998].

- Objectivos mais claros para os investimentos de TI - maior diferenciação no apoio a novas formas de fazer negócio, melhorando a flexibilidade ou facilitando a comunicação com os clientes;
- Maior envolvimento dos líderes seniores na governança de TI - maior participação dos CIOs, chefes de negócios e das unidades de negócio (quem possa descrever fielmente o regime da governança de TI);
- Maior estabilidade na sua governança (isto é, menos mudanças de ano para ano);
- Bom funcionamento de processos formais de excepção.

Abordagem centralizadas e descentralizadas

Num modelo de governança de TI centralizado, os executivos seniores e corporativos têm poder de decisão nos investimentos de TI, que incluem [Peterson, 2004]:

- Aplicações de negócio: as prioridades de aplicações de TI e de planeamento, orçamento e entrega/manutenção de serviços de aplicação específicos de negócio;
- Serviços Partilhados /Arquitectura: o conhecimento dos processos de negócio e das funções com a capacidade da infraestrutura de TI, dos sistemas de TI desenvolvidos ao longo do seu ciclo de vida e os padrões de arquitectura de TI para os dados, os aplicativos e a tecnologia;
- Componentes de tecnologia /Plataforma: plataformas de hardware/software, redes e as normas para aquisição e implantação de recursos de TI.

Quando todo o poder de decisão de TI é atribuído a diferentes linhas de negócio, separadas por divisões de negócio, ou unidades estratégicas de negócio, a estrutura é descrita como um modelo completamente descentralizado de governança de TI [Peterson, 2004].

Em geral, se todos os restantes factores permanecerem inalterados, a centralização leva a uma maior especialização, economias de escala, consistência, e controlos normalizados, enquanto que a descentralização permite o controlo do negócio, uma sensação de propriedade do negócio, e proporciona uma maior capacidade de resposta e flexibilidade para as necessidades do negócio (ver tabela 3.22). No entanto, excesso de flexibilidade na descentralização pode conduzir a padrões variáveis que podem, em última instância, resultar em menor flexibilidade. Por outro lado, a especialização na centralização incorre em riscos estratégicos específicos, devido à racionalidade limitada e à sobrecarga de informação [Peterson, 2004].

	Governança de TI centralizada	Governança de TI descentralizada	Governança de TI federal
Sinergia ²³ de TI	+	-	+
Normalização da TI	+	-	+
Especialização da TI	+	-	+
Tempo de resposta do negócio	-	+	+
Dono do negócio	-	+	+
Flexibilidade do negócio	-	+	+

Tabela 3.22: Trade-offs e o melhor dos dois modelos (adaptado de [Brown and Magill, 1998] e [Rockart, 1996])

Uma visão política da governança de TI sugere, contudo, que o debate sobre centralização versus descentralização é principalmente utilizado para a prossecução dos objectivos de *stakeholders* específicos, em certas formas que não poderiam ajudar a empresa a cumprir objectivos [Simon and Barnard, 1961] [Cyert, 1963]. Recordando a lenda dos seis homens cegos e o elefante²⁴: dentro da empresa há diferenças importantes entre os *stakeholders*, levando à presença de conflitos e de desacordos sobre a definição de metas e de recursos estratégicos (incluindo a TI). Estes *stakeholders* representam diferentes grupos ou indivíduos que influenciam, e são afectados por, decisões relativas à TI.

As lutas de poder, turbulência política, e confrontos culturais são endémicos para a governança de TI, e a seguinte questão é ainda mais "afinal, de que maneira é que vai ser", em vez de "qual é o melhor caminho". Apesar de não ter sido sempre explicitamente reconhecida, documentada ou entendida, a resolução de conflitos e a construção de coligações são os processos-chave para o funcionamento eficaz de qualquer tipo de modelo de governança de TI [Peterson, 2001].

Além da política, o potencial risco no ambiente de negócio contemporâneo é que, quer a centralização, quer a descentralização, "fixam" a organização numa postura rígida. O desafio é, portanto, equilibrar os benefícios (custos e riscos) da centralização e da descentralização da governança de TI [Peterson, 2004].

Ao longo da última década muitas das organizações, se não a maioria, têm definido que para alcançar "o melhor de dois mundos" é necessário adoptar o modelo federal da governança de TI [Peterson, 2001] [Sambamurthy e Zmud, 1999] [Weill, 2004].

²³ Sinergia - cooperação; convergência das partes de um todo que concorrem para um mesmo resultado.

²⁴ Conta uma lenda hindu que seis homens cegos foram colocados junto de um elefante e foi-lhes pedido para o descrever: um tocou na perna e disse que o elefante era como uma "árvore"; outro agarrado à cauda disse que era como uma "corda"; outro sentiu o lombo e disse que era como uma "parede"; outro sentiu os grandes dentes e disse que era como uma "lança"; um outro ainda disse que era como um "leque" ao sentir a orelha; por último o que se agarrou à tromba respondeu logo que era como uma "cobra". Então, estes homens do Hindustão discutiram alto e durante muito tempo, cada um mantendo a sua opinião de um modo excessivamente forte e inflexível. Todos os seis homens estão correctos. O motivo pelo qual, cada um disse coisas diferentes, é, porque cada um tocou numa parte diferente do elefante. Então, de facto, o elefante possui todas as características mencionadas. É assim que os homens se comportam perante a verdade. Conhecem apenas uma parte, mas pensam que é o todo! [Godfrey, 1963]

Sob o modelo federal da governança de TI, as decisões de infraestrutura de TI - fornecimento da tecnologia - são centralizadas e as decisões de aplicação informática – uso da tecnologia - são descentralizadas [Brown and Magill, 1998]. Contudo, o modelo federal não tem uma estrutura monolítica. Existem diferentes padrões de diferenciação dentro do modelo federal da governança de TI: modelos federais centrados na TI e modelos federais centrados no negócio.

- Num modelo federal, centrado na TI, o corpo executivo de TI é responsável pelas redes e pelas decisões de desenvolvimento das infraestruturas de TI, e a gestão de TI é responsável pelas decisões de serviços de aplicações de negócio.
- Num modelo federal, centrado no negócio, os executivos de negócio desempenham um papel de liderança nas decisões de aplicação do negócio.

A chave para compreender a diferença entre os modelos federais centrados na TI ou no negócio, é o nível de envolvimento e participação na tomada de decisões dos executivos de negócio na TI [Peterson, 2004].

Embora tradicionalmente focada, quer na eficiência, quer na flexibilidade, muitas vezes numa forma sequencial (por vezes levada a um contínuo balançar entre a centralização e a descentralização), hoje, a governança de TI enfrenta a dupla exigência de flexibilidade e agilidade, por um lado, e a eficiência e a normalização por outro. Os executivos de TI e de negócio têm vindo a reconhecer que precisam de cumprir a procura personalizada de produtos e serviços de TI de alta qualidade e que necessitam de normalizar e controlar custos e tempo, a fim de satisfazer as necessidades da empresa de uma forma eficiente, fiável e eficaz [Peterson, 2004].

A Governança de TI contemporânea não se pode permitir a incidir sobre o serviço de infraestrutura nos custos de integração de soluções, ou vice-versa. Ademais, é difícil, senão impossível, alcançar inovação estratégica sem alguma base no desempenho de serviços de infraestruturas e de integração de soluções [Peterson, 2004].

3.7 Planear e avaliar a governança de TI

A Governança eficaz exige a harmonização de dois tipos: "o quê" e "como".

"O quê" inclui objectivos da empresa, arquétipos de governança de TI e medidas de desempenho. Os objectivos da empresa podem ser expressos como controladores financeiros e máximas de negócios [Broadbent and Weill, 1997]. Os arquétipos de governança de TI são sobre a escolha dos seis arquétipos de governança – monarquia do negócio, monarquia de TI, feudal, federal, duopólio e anarquia – que estão mapeados para os direitos de decisão de cada um dos cinco domínios de decisão de TI - princípios de TI, arquitectura de TI, infraestrutura de TI, necessidades de aplicações de negócio e investimentos de TI. Os objectivos de desempenho são os alvos e os quadros temporais para a entrega dos objectivos da empresa.

O "como" inclui comportamentos de TI desejáveis que fluem dos objectivos da empresa, dos mecanismos de governança de TI em vigor e das métricas de TI e responsabilidades [Weill and Woodham, 2002].

Para que uma empresa possa analisar o desempenho da sua governança de TI pode começar por utilizar um simples teste de cinco questões, ilustrado na tabela 3.23. Cada resposta a essas questões é pontuada de 0 a 3 para avaliar quão verdadeiros são, cada um dos seguintes itens:

Indicadores da eficácia da governança de TI	Feito pobremente / nada feito (peso 0)	Algo feito pobremente (peso 1)	Algo feito bem (peso 2)	Bem feito/ completamente (peso 3)
1. São ponderadas cuidadosamente as variações nos estilos utilizados para os diferentes domínios de decisão de TI				
2. Os executivos de alto nível estão envolvidos na governança de TI				
3. Os executivos de alto nível podem descrever com precisão a governança de TI				
4. Existem comissões executivas mais ou menos formais - Caso existam, sustentam a governança de TI				
5. É realizada a monitorização: - do valor de negócio de TI - do consumo de projectos - de recursos				
Total				

Tabela 3.23: Avaliação da eficiência da governança de TI (adaptado de [Weill and Woodham, 2002] e [Broadbent and Weill, 1998])

A contagem dos indicadores é a seguinte:

- 0-3** Sem governança eficaz de TI
- 4-7** Nível baixo de governança de TI
- 8-11** Governança de TI madura
- 12-15** Desempenho de topo.

Na avaliação ou implementação da governança devem considerar-se os factores críticos de sucesso descritos na secção seguinte.

3.8 Recomendações para a governança de TI

É interessante questionar como é que uma empresa pode harmonizar o seu perfil de governança de TI com os objectivos da empresa. Para responder a esta questão enumeramos, seguidamente, seis factores críticos de sucesso necessários para uma governança eficaz (adaptado de [Weill and Woodham, 2002] e [Broadbent and Weill, 1998]):

- 1) A governança de TI deve ser cuidadosamente e activamente concebida. A gestão de executivos deve ser envolvida para que seja eficaz.

- 2) Sem transparência, não há confiança. A transparência deve ser construída para que a governança de TI confie nos processos.
- 3) Devemos reconhecer que mudar a governança pode levar meses. Devemos fazer as alterações apenas quando comportamentos convenientes mudam acentuadamente.
- 4) Devemos educar os executivos e os gestores sobre o porquê da importância da governança ser um constante desafio e exigência. Bons comportamentos devem ser reforçados e comportamentos inadequados redireccionados.
- 5) Um enfoque acentuado num número limitado de objectivos, comportamentos e métricas é necessário. A boa governança exige escolhas. Podemos otimizar em múltiplas opções.
- 6) Devem existir excepções claras no tratamento dos processos, pois as excepções são a forma como as empresas aprendem. As excepções encorajam debates mas fazem prosperar rapidamente.

Em conclusão, nós acreditamos que governar uma empresa é uma tarefa apenas para a gestão de topo. É necessária, mas não suficiente, e por vezes não é suficientemente poderosa para o resto da empresa [Weill, 2004]. Executivos de topo, incluindo os CIOs, também têm de governar. A boa governança:

- Assegura que os grupos certos estão a fazer as principais/importantes decisões de TI a fim de que essas decisões alcancem os objectivos e os comportamentos desejáveis da empresa;
- Clarifica quem pode tomar decisões e como eles são responsáveis pelas metas da empresa;
- Dá poder aos gestores da empresa para tomarem decisões sem solicitarem aprovação adicional da gestão sénior (quando permitido pela *framework* de governança).

Uma prova de boa governança de TI é saber se o CIO pode deixar a empresa durante 1 ou 2 meses para desempenhar uma diligência sobre uma potencial fusão e aquisição. Enquanto o CIO está afastado e não está a liderar, interessa questionar se a governança irá trabalhar eficazmente ou se requer a liderança do CIO para funcionar.

Capítulo 4

Avaliar a Governança de TI

Conteúdo

4.1 QUESTIONÁRIO SOBRE A GOVERNANÇA DE TI.....	58
4.2 COMO AS EMPRESAS GOVERNAM A TI: DECISÕES-CHAVE E ARQUÉTIPOS DE TI.....	59
4.3 FACTORES DE VARIAÇÃO NOS PERFIS DA GOVERNANÇA DE TI.....	63
4.4 PERFIS DA GOVERNANÇA DE TI.....	64
4.5 ABORDAGENS DA GOVERNANÇA DE TI.....	66

Este capítulo relata detalhadamente a abordagem adoptada para determinar os arquétipos e os perfis de governança de TI das empresas constantes da amostra. É descrito o processo de elaboração, recolha e análise do questionário sobre a governança de TI.

4. Avaliar a Governança de TI

4.1 Questionário sobre a Governança de TI

Para determinarmos o perfil de governança de TI das empresas portuguesas adoptámos a seguinte metodologia:

- Adaptámos um questionário sobre como avaliar a governança de TI numa organização;
- Seleccionámos uma amostra de empresas na área de TI e distribuímos os questionários;
- Analisámos os resultados obtidos e determinámos o perfil das empresas.

Este questionário foi a ferramenta necessária para podermos fazer o levantamento do grau de maturidade e do perfil de governança de uma organização.

O processo de elaboração do questionário para avaliar a governança de TI numa organização baseou-se em trabalho de investigação desenvolvido na *Sloan School of Management* do MIT [Hoffmann and Weill, 2007], que foi resumido no capítulo anterior.

Este inquérito, tem duração estimada de preenchimento de 20 minutos, contém 12 perguntas de resposta rápida e está dividido em 3 secções: Elementos de Governança de TI, Desempenho da Governança de TI e Contexto Organizacional. O inquérito encontra-se no Anexo A – secção a)).

O processo de recolha de dados foi demorado e atribulado. Começámos por distribuir os questionários em papel em alguns eventos e workshops nacionais. Como o número de respostas obtido foi bastante reduzido, decidimos disponibilizar o questionário online através do Moodle²⁵ <http://moodle.fct.unl.pt/>. Divulgámos o site através do itSMF Portugal – *IT Service Management Fórum* (<http://www.itsmf.pt/>) e conseguimos obter 17 respostas online. Todavia, foi graças ao itSMF Portugal que conseguimos obter um alargamento considerável da amostra com mais 35 inquéritos durante a sua conferência anual (<http://www.itsmf.pt/Portals/0/Conferencias/2009/Destaque.pdf>) que decorreu nas instalações da Reitoria da UNL a 27 de Maio de 2009. O itSMF Portugal convidou responsáveis de topo de grandes empresas nacionais para nos darem o testemunho do valor que as TI acrescentam aos seus negócios e a criticidade que a qualidade da sua gestão representa para o sucesso e a competitividade das suas empresas.

Apelámos sempre aos inquiridos que preenchessem o questionário tão completamente quanto possível, apelando à veracidade e sinceridade na sua resposta. O anonimato e a confidencialidade dos dados individuais fornecidos foram absolutamente garantidos.

²⁵ O Moodle é um sistema *open-source* de gestão da aprendizagem e de trabalho colaborativo que permite a produção expedita de sítios *web*. Dado que tem facilidades de construção de inquéritos, pôde ser usado para este fim no âmbito desta dissertação [http://docs.moodle.org/pt/Sobre_o_Moodle] [Wikipédia, Moodle].

O processo de análise e interpretação dos dados também se encontra no anexo A, na secção b) e c), respectivamente. Ao longo deste capítulo relatam-se os resultados obtidos. Com estes resultados pudemos analisar e mapear os tipos de decisões existentes numa empresa, os arquétipos para a tomada dessas decisões e como é que essas decisões são tomadas e monitorizadas, através de um conjunto de mecanismos de governança.

Temos como objectivo apresentar publicamente os resultados globais num workshop do itSMF Portugal a anunciar.

4.2 Como as Empresas Governam a TI: Decisões-chave e Arquétipos de TI

Estudámos a governança de TI em várias empresas com múltiplas unidades de negócios (ver figura 4.7 para mais detalhes).

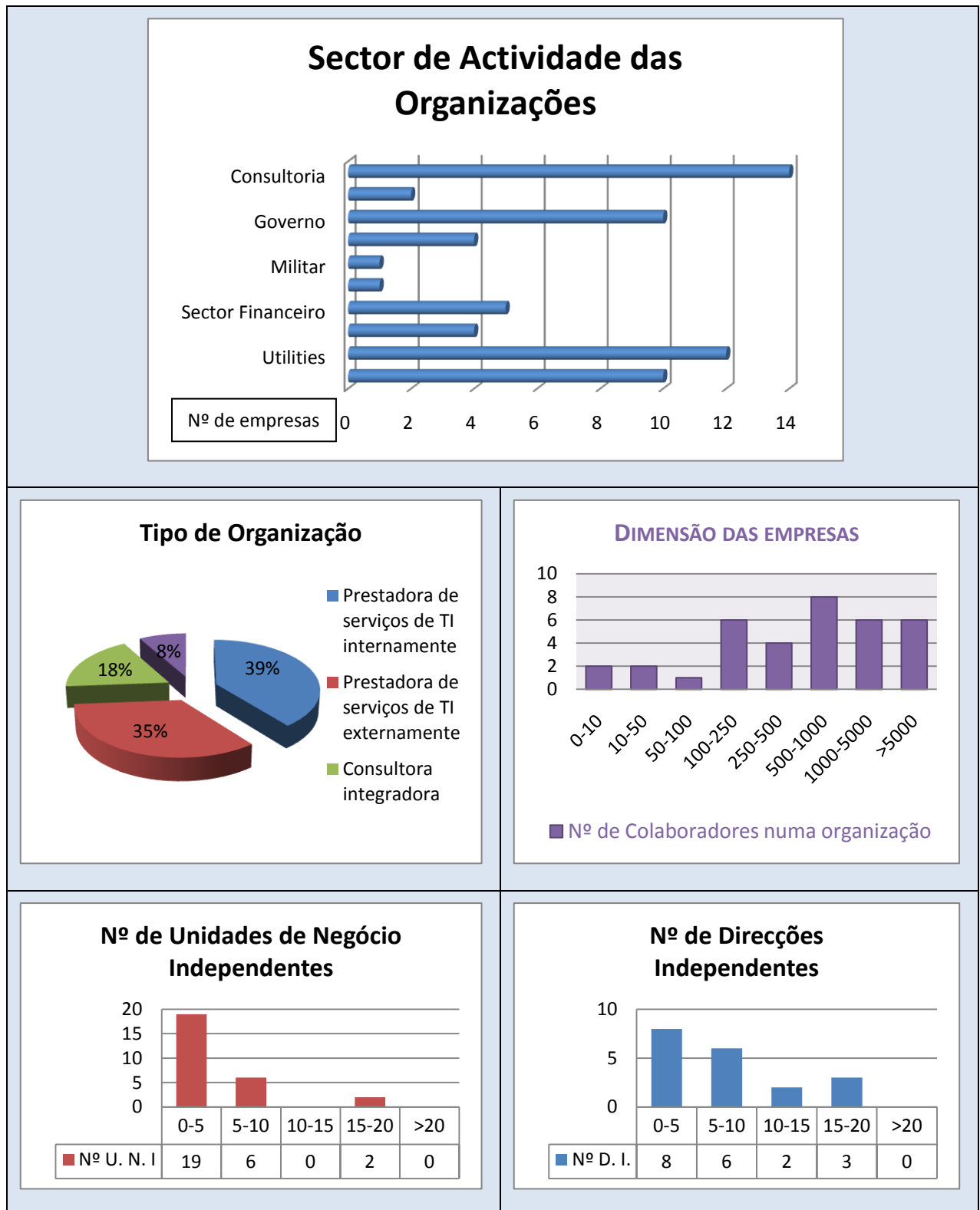


Figura 4.7: Alguns dados sobre a amostra de empresas

Para a empresa como um todo, estudámos, tanto quem detém os direitos de decisão, como quem detém os direitos de contribuir para a decisão de cada um dos cinco domínios de decisões. Depois, categorizámos a abordagem da empresa pelos arquétipos.

A tabela 4.24 mostra dois resultados:

1. Os números nas células mostram a percentagem das empresas que utilizam cada arquétipo de governança para cada decisão.
2. Para ser mais legível, as células listadas e sombreadas a laranja indicam os direitos de decisão e os direitos mais comuns de contribuir para cada decisão (a maior percentagem numa coluna).


Domínio Arquétipo	Princípios de TI	Arquitectura de TI	Infra-estrutura da TI	Necessidades de aplicações de negócio	Investimentos de TI
Monarquia do negócio	65,7 %	28,6 %	34,2 %	25,7 %	62,9 %
Monarquia de TI	0,0 %	28,6 %	14,3 %	14,3 %	5,7 %
Feudal	0,0 %	17,1 %	11,4 %	11,4 %	5,7 %
Federal	2,9 %	2,9 %	8,6 %	11,4 %	2,9 %
Duopólio	31,4 %	22,8 %	28,6 %	34,3 %	22,8 %
Anarquia	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %
Não aplicável	0,0 %	0,0 %	2,9 %	2,9 %	0,0 %
Total²⁶	100,0 %	100,0 %	97,1 %	97,1 %	100,0 %
Os números em cada célula são a percentagens dos 35 questionários.			 Padrão mais comum para todas as empresas		

Tabela 4.24: Arquétipos de governança usados para diferentes tipos de decisões

Princípios de TI – Os princípios de TI definem o papel estratégico para a TI em toda a empresa, e são decididos de duas formas. Cerca de 66% das empresas utilizam uma abordagem “monarquia do negócio” e 31% utilizam uma abordagem “duopólio”.

A “monarquia do negócio” é utilizada pela maioria das empresas, porque os executivos seniores permitem à TI moldar os princípios de negócio, reflectindo as capacidades da TI já existentes ou em desenvolvimento nesses princípios.

A abordagem “duopólio” também é regularmente utilizada nos princípios de TI, porque assegura o compromisso da unidade de TI com os princípios do negócio.

Arquitectura de TI – Aproximadamente 29% das empresas conta com a “monarquia do negócio” e com a “monarquia de TI” para tomar as decisões da arquitectura de TI, e 23% conta com “duopólio”. Segundo o nosso estudo, estes 3 arquétipos diferem dependendo da dimensão da empresa. Quando falamos em grandes empresas (> de 250 colaboradores), quem toma as decisões da arquitectura de TI é a “monarquia de TI”, sugerindo que os executivos de negócio seniores vêm a arquitectura mais como uma questão técnica do que como uma questão estratégica de negócio. O papel dominante da tomada de decisão de TI nas decisões da arquitectura sugere que os gestores de negócio se sintam incompetentes, desinteressados ou desnecessários - eles confiam que os profissionais de TI conseguem

²⁶ O total, tanto para o domínio de “infraestrutura de TI”, como para o domínio de “necessidades de aplicações de negócio” não é de 100% porque, houve um inquirido que respondeu “não aplicável” a estes dois domínios de decisão. Este tipo de resposta não contou para o total.

traduzir os princípios de TI numa arquitectura. Os profissionais de TI sentem-se, normalmente, bastante confortáveis nesta área, tendo a responsabilidade pelas decisões da arquitectura.

Quando falamos em médias empresas (entre 50 e 250 colaboradores), quem toma as decisões da arquitectura de TI é a “monarquia de negócio”, pois os executivos seniores têm a percepção de que devem assumir a liderança para garantir que a TI está alinhada com as estratégias de negócio.

Quando falamos em pequenas empresa (menos de 50 colaboradores), quem toma as decisões da arquitectura de TI é o “duopólio”, ou seja, são os directores em conjunto com as chefias que têm responsabilidade pelas decisões da arquitectura de TI. Trabalhar em parceria com os líderes de TI no processo de decisão estabelece expectativas realistas para a TI e força a clarificação das estratégias de negócios.

A arquitectura de TI é a única decisão onde um número considerável de empresas (17%) utilizam um modelo feudal - unidades de negócio independentes tomam as suas próprias decisões.

Infraestrutura da TI – Como na arquitectura de TI, a infraestrutura de TI também difere dos arquétipos, dependendo da dimensão da empresa. Cerca de 34% das pequenas e médias empresas utilizam a “monarquia de negócio” e cerca de 29% das grandes empresas utilizam “duopólio”.

A infraestrutura, se for bem projectada, possibilita a utilização de aplicações logo que necessárias. Pobrememente projectada, a infraestrutura é uma barreira frustrante para as unidades de negócio. Envolvendo líderes de negócio nestas decisões de infraestruturas (“duopólio”) torna-as mais eficazes, reduzindo o risco da TI. O “duopólio” eficaz desenha infraestruturas para antecipar e apoiar os requisitos de aplicação de unidades de negócio. As pequenas e médias empresas utilizam a “monarquia de negócio”, porque a sua pequena dimensão faz com que haja pouca segregação de funções. Muitas vezes, estas empresas não têm unidades de negócio, sendo o director o único líder de negócio.

Necessidade de aplicações de negócio – Assim como acontece no domínio de decisão sobre a infraestrutura de TI, também na necessidade de aplicações de negócio, cerca de 26% das pequenas e médias empresas utilizam a “monarquia de negócio” e cerca de 34% das grandes empresas utilizam “duopólio”.

A maioria das empresas escolhe o modelo “duopólio” para tomar as decisões relativamente às necessidades de aplicações de negócio, para que os objectivos da empresa estejam incluídos no processo de implantação de aplicações locais. Usando o modelo “duopólio” uma empresa, por exemplo, pode decidir se quer replicar ou personalizar o software a nível da empresa para o nível local.

Por exemplo, uma empresa adquire um sistema para toda a empresa. Mas, à excepção de um pequeno conjunto de definições de dados (por exemplo, dados financeiros), a aplicação não foi padronizada em todas as suas unidades de negócio. Uma solução, seria a equipa de desenvolvimento desenvolver um

modelo, e ajudar as equipas locais a configurar o sistema para atender às suas necessidades únicas. Este acordo permite partilhar conhecimentos em toda a empresa, mas considera os benefícios de personalização local mais importantes do que a padronização global.

Investimentos de TI – As empresas tomam decisões de investimento de TI utilizando dois arquétipos: aproximadamente 63% usa “monarquia do negócio” e 23% usa “duopólio”. Estes dois arquétipos oferecem modelos diferentes de como as empresas pretendem maximizar o valor dos seus investimentos de TI. A “monarquia do negócio” faz tomadas de decisão *top-down* para implementar uma alteração na estratégia. Um modelo “duopólio” é normalmente utilizado quando, por exemplo, decisões bilaterais (entre a TI e cada unidade de negócio) satisfazem as necessidades individuais das unidades de negócio, enquanto compartilham serviços de TI. Num “duopólio”, o grupo de TI desempenha um importante papel de intermediário, e normalmente não é necessário consenso entre as unidades de negócio.

Apenas 6% das empresas coloca as decisões de investimento de TI nas mãos de profissionais de TI, o que reflecte a crescente sensibilização/consciencialização de que tais decisões envolvem compromissos do negócio. Os decisores de negócio apercebem-se que precisam de determinar quais são os processos de negócio que irão receber apoio da TI e quais não receberão.

4.3 Factores de Variação nos Perfis da Governança de TI







O nosso estudo revela alguns perfis da governança de TI. Por exemplo, são poucas as empresas que governam a TI utilizando as abordagens anarquia ou feudal. Observou-se que as variações nos perfis de governança resultam principalmente dos cinco factores seguintes [Weill, 2004]:

1. Objectivos estratégicos e de desempenho: a governança eficaz tenta reforçar comportamentos desejáveis específicos para atingir os objectivos estratégicos e de desempenho da empresa.
2. Estrutura organizacional: tradicionalmente, as empresas têm invocado a estrutura organizacional para alinhar a tomada de decisão com os objectivos e estratégias da empresa. No entanto, as empresas tentam abordar os objectivos concorrentes, a expansão geográfica, as mudanças rápidas e a competição intensa, mostrando que as estruturas organizacionais se têm revelado um suporte insuficiente para a estratégia. Agora, as empresas modelam a governança, em parte, para compensar as limitações de estrutura e para exigir menos alterações na estrutura à medida que o negócio muda.
3. Experiência de governança: muitas empresas são relativamente precoces, na curva de aprendizagem da governança de TI. Como a sua experiência aumenta, alteram a sua governança de TI.
4. Dimensão e diversidade: uma vez que as empresas crescem e se diversificam, tanto geograficamente como organizacionalmente, introduzem-se objectivos concorrentes, e até mesmo, conflituosos. A governança de TI reflecte essas alterações.

5. Indústria e diferenças regionais: A indústria e as diferenças regionais criam pressões únicas nas empresas que são reflectidas na sua governança de TI. As culturas da tomada de decisão variam consideravelmente nas diferentes regiões do mundo, muitas vezes complicando a governança em empresas globais.

4.4 Perfis da Governança de TI

Nós encontramos uma variação significativa nos perfis de governança de TI entre as empresas inquiridas. Calculámos os seus perfis de governança de TI, através da pergunta 1 do grupo A do questionário, que se encontra no anexo A, secção a). Com base na figura 3.6, na tabela 3.19, na tabela 4.24, e no valor médio dos 52 inquéritos respondidos, obtivemos 2 perfis de governança para as empresas portuguesas, tal como representado na tabela 4.25.

Domínio Arquétipo	Princípios de TI	Arquitectura de TI	Infraestrutura de TI	Necessidades de aplicações de negócio	Investimentos de TI
Monarquia do negócio					
Monarquia de TI					
Feudal					
Federal					
Duopólio					
Anarquia					

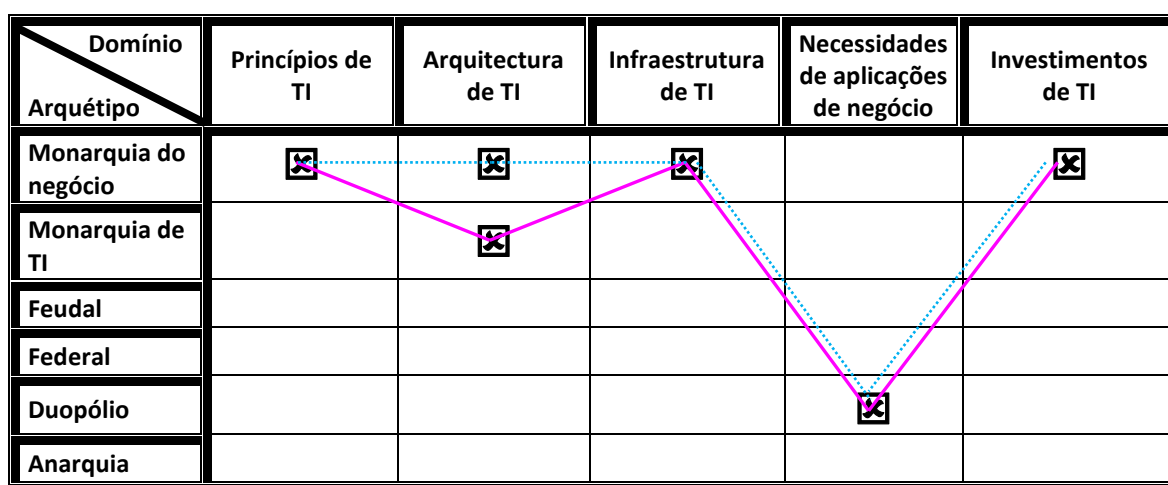


Tabela 4.25: Perfil de governança de TI das empresas portuguesas

Os dois perfis são quase coincidentes. A única diferença situa-se ao nível da decisão sobre “Arquitectura de TI”, onde tanto podem ser os directores a decidirem (Monarquia do negócio) como as chefias a decidirem (Monarquia de TI), como podemos constatar na tabela 4.25.

Façamos agora uma comparação com a realidade observada pelos investigadores americanos do MIT²⁷, que é descrita no capítulo 3. A partir da figura 3.6, podemos concluir que o perfil a tracejado e a azul, na tabela 4.25, está muito próximo do perfil 3, diferindo apenas no domínio de decisão - necessidades de aplicação do negócio, que no nosso estudo não é federal, mas duopólio. O perfil a rosa, na tabela 4.25, está tão próximo do perfil 2 como do perfil 3. No perfil 2 difere nos domínios de decisão – princípios de TI e infraestrutura de TI. No perfil 3 difere nos domínios de decisão – arquitectura de TI e necessidades de aplicação do negócio, como é apresentado na tabela 4.26.

²⁷ Os investigadores americanos do MIT estudaram a governança de TI em 256 empresas, com várias unidades de negócios, com e sem fins lucrativos e em 23 países da América, da Europa e da Ásia [Weill, 2004].

	Os melhores três tipos de desempenho [Weill, 2004]			Nosso Estudo		
	Linha tracejado Perfil 3	Linha continua Perfil 2	Linha continua Perfil 3	Linha tracejado Perfil 3	Linha continua Perfil 2	Linha continua Perfil 3
Princípios de TI	Monarquia do negócio	Duopólio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio
Arquitectura de TI	Monarquia do negócio	Monarquia de TI	Monarquia do negócio	Monarquia do negócio	Monarquia de TI	Monarquia de TI
Infraestrutura de TI	Monarquia do negócio	Monarquia de TI	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio
Necessidades de aplicações de negócio	Federal	Duopólio	Federal	Duopólio	Duopólio	Duopólio
Investimentos de TI	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio

Tabela 4.26: Comparação do Perfil de governança de TI das empresas constantes da amostra com os três melhores tipos de desempenho identificados em [Weill, 2004]

Concluimos, então que o perfil de governança de TI das empresas portuguesas situa-se entre os 60% (linha rosa) e os 80% (linha azul) de dois dos três perfis mais eficazes, medidos pelo desempenho da governança de TI. Este resultado revela-nos que as empresas portuguesas estão a adoptar estratégias focadas, têm objectivos claros para os investimentos de TI e que os líderes seniores têm envolvimento na governança de TI.

Como observámos nas secções 3.3 “Como as empresas governam a TI: Decisões-chave e Arquétipos de TI” e na 3.5 “Perfis da governança de TI”, que um dos factores que afecta os perfis de governança de TI é a dimensão e a diversidade das empresas, decidimos fazer o estudo do arquétipo de governança de TI separadamente para pequenas (menos de 50 colaboradores), médias (entre 50 e 250 colaboradores) e grandes (mais de 250 colaboradores) empresas²⁸ (tabela 4.27).
















Domínio Arquétipo	Princípios de TI	Arquitectura de TI	Infraestrutura de TI	Necessidades de aplicações do negócio	Investimentos de TI
Monarquia do negócio	  		 	 	  
Monarquia de TI					
Feudal					
Federal					
Duopólio					
Anarquia					

Tabela 4.27: Perfil de governança de TI das pequenas, médias e grandes empresas constantes da amostra

²⁸ Para mais informações sobre a definição de pequenas, médias e grandes empresas, consultar a *European Commission* (2003-05-06), *Recommendation 2003/361/EC: SME Definition*, http://ec.europa.eu/enterprise/enterprise_policy/sme_definition/index_en.htm

A partir da figura 3.6 podemos concluir que o perfil das pequenas empresas está próximo do perfil 3, diferindo nos domínios de decisão – arquitectura de TI e necessidades de aplicação do negócio. O perfil das médias empresas encontra-se ainda mais próximo do perfil 3, diferindo apenas no domínio de decisão – necessidades de aplicação do negócio. Por fim, o perfil das grandes empresas está próximo do perfil 2, diferindo nos domínios de decisão – princípios de TI e infraestrutura de TI, como é apresentado na tabela 4.28.

	Os melhores três tipos de desempenho [Weill, 2004]			Nosso Estudo		
	Pequenas perfil 3	Médias perfil 3	Grandes perfil 2	Pequenas perfil 3	Médias perfil 3	Grandes perfil 2
Princípios de TI	Monarquia do negócio	Monarquia do negócio	Duopólio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio
Arquitectura de TI	Monarquia do negócio	Monarquia do negócio	Monarquia de TI	Duopólio	Monarquia do negócio	Monarquia de TI
Infra-estrutura de TI	Monarquia do negócio	Monarquia do negócio	Monarquia de TI	Monarquia do negócio	Monarquia do negócio	Duopólio
Necessidades de aplicações de negócio	Federal	Federal	Duopólio	Monarquia do negócio	Monarquia do negócio	Duopólio
Investimentos de TI	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio	Monarquia do negócio

Tabela 4.28: Comparação do perfil de governança de TI das pequenas, médias e grandes empresas constantes da amostra com os três melhores tipos de desempenho identificados em [Weill, 2004]

Como já pudémos concluir através da tabela 4.27, aqui voltamos a confirmar, de uma forma menos geral, que o perfil de governança de TI das pequenas e grandes empresas portuguesas situa-se nos 60% (linha rosa e amarela) dos respectivos perfis (perfil 3 e 2) mais eficazes e o das médias empresas portuguesas situa-se nos 80% (linha azul) do perfil 3.

Temos contudo sérias reservas quanto à validade dos resultados, tanto das pequenas, como das médias empresas, pois tínhamos amostras muito reduzidas, de 4 e 7 inquéritos respondidos respectivamente.

4.5 Abordagens da governança de TI

Alcançar um desempenho de governança elevado significa que a governança de TI da empresa foi bem sucedida na influência das medidas desejadas de sucesso.

Avaliar o desempenho da governança de TI

Segundo Peter Weill e Jeanne Ross, o desempenho de governança de uma empresa ou de uma unidade de negócio é avaliado, calculando a eficácia da governança de TI na prestação de quatro objectivos ponderados pela sua importância para a empresa [Weill and Ross, 2004]:

- Boa rentabilização do investimento em TI;

- Uso eficaz da TI para melhorar a utilização dos recursos (humanos e materiais) da organização;
- Uso eficaz da TI para o crescimento do negócio;
- Uso eficaz da TI para conceber maior flexibilidade ao negócio.

Quando os gestores seniores avaliam o desempenho da governança, primeiro identificam a importância relativa de cada um dos 4 factores na sua empresa. Seguidamente classificam o desempenho da empresa em cada factor.

A medida de desempenho da governança foi criada com base nas perguntas 1 e 2 do grupo B do questionário, que se encontra no anexo A, na secção a). A pergunta 1 avalia a importância de um resultado particular e a pergunta 2 avalia o quão bem a governança de TI contribui para esse resultado. Uma vez que nem todas as empresas classificaram o resultado com a mesma importância, as respostas à primeira questão foram usadas para ponderar as respostas à segunda questão. Portanto, matematicamente temos:

Desempenho da governança

$$= \frac{\sum_{n=1}^4 (\text{importância do resultado } \{i.e. \text{ Questão 1}\} * \text{influência da governança de TI } \{i.e. \text{ Questão 2}\}) * 100}{\sum_{n=1}^4 (5(\text{importância do resultado}))}$$

Dado existirem quatro objectivos, a pontuação máxima é de 100 e a mínima é de 20. O desempenho da governança de TI para a média dos 52 inquéritos, para as pequenas, médias e grandes empresas está representado na figura 4.8. Reforçamos, que não devemos considerar como representativos os resultados, tanto das pequenas empresas, como das médias empresas, pois tínhamos amostras muito reduzidas.

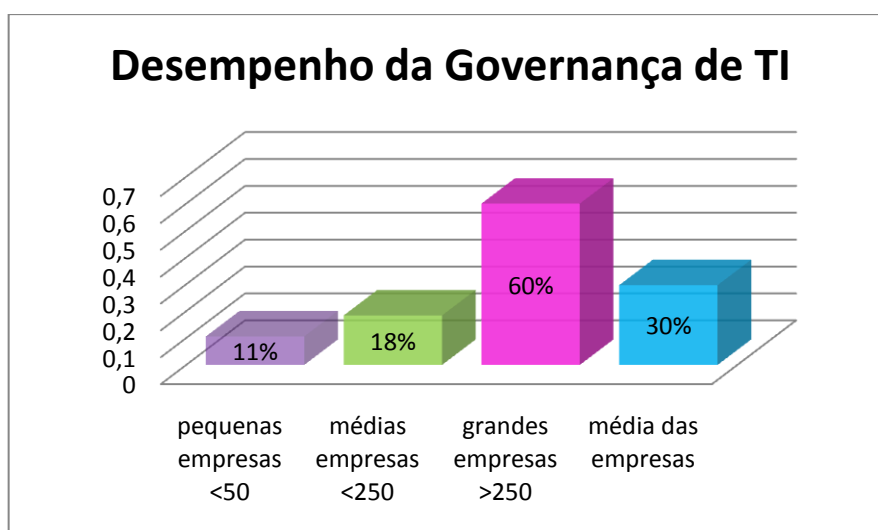


Figura 4.8: Desempenho da governança de TI para a média dos 52 inquéritos, para as pequenas, médias e grandes empresas constantes da amostra

Abordagem centralizada

Na nossa pesquisa, as empresas tendem a ser centralizadas na sua abordagem à governança de TI. Conforme mostrado na tabela 3.21, as estratégias destas empresas enfatizam operações eficientes frequentemente centradas em medidas do custo dos processos de negócio e da rentabilidade. Por conseguinte, o comportamento desejável da TI personifica um elevado grau de normalização na prossecução de baixos custos de negócio. Os mecanismos chave da governança incluem o Conselho de Administração para a tomada de decisões, os processos centralizados para a concepção da arquitectura, os processos de decisão de investimento de TI a nível da empresa e avaliações formais após implementação de projectos relacionados com a TI.

A governança mais centralizada permite a normalização, tanto dos processos de negócio, como da TI [Weill and Ross, 2004].

Abordagem descentralizada

Na nossa pesquisa, são muito poucas as empresas que seguem uma abordagem descentralizada. Em relação ao crescimento, os gestores de topo focam-se mais na inovação e no tempo de mercado. Estas empresas insistem em prestar contas (ter responsabilidades) localmente. Elas medem o sucesso através do crescimento das receitas. Ainda no crescimento, os gestores de topo minimizam constrangimentos na criatividade e na autonomia²⁹ da unidade de negócio, estabelecendo poucas, se algumas, tecnologias a nível da empresa e normas nos processos de negócio. Assim, exigem poucos mecanismos de governança, muitas vezes confiando apenas num processo de investimento que identifica projectos estratégicos com prioridade alta e gere riscos. “Que resultados” é a pergunta às necessidades dos clientes locais, ao alto crescimento e à pouca normalização do nível das empresas [Weill and Ross, 2004].

Abordagem híbrida

As empresas apostam na utilização de activos na tentativa de equilibrar o contraste entre a governança para a rentabilidade e a governança para o crescimento das receitas e para a inovação. Estas empresas focam-se na utilização de serviços partilhados para alcançar tanto o tempo de resposta do cliente, como a economia de escala³⁰, como ambos. Os seus princípios de TI enfatizam a partilha e a reutilização de processo, sistema, tecnologia e módulos de dados.

Tipicamente, os líderes, na utilização de activos, dependem dos arquétipos de governança duopólio e federal. Eles introduzem mecanismos de governança para abordar as tensões entre o controlo a nível da empresa e local. Os mecanismos incluem o relacionamento de gestores de negócio / TI, acordos de

²⁹ Autonomia - qualidade de um indivíduo tomar as suas próprias decisões, com base na sua razão individual; liberdade moral ou intelectual.

³⁰ Economia de escala - um custo mais baixo por unidade produzida, alcançado através de uma produção em grande escala. Uma grande operação de alimentação de gado poderia ser capaz de beneficiar de economias, tais como custos mais baixos por unidade de alimentação, incremento da mecanização e custos de veterinário mais baixos por unidade [Wikipédia, Economies of scale].

nível de serviço e de recuperação de custos de TI, equipas de liderança de TI que compreendem os representantes de TI da unidade de negócio e equipas de processos com membros de TI. A utilização de activos exige uma abordagem híbrida para a governança, misturando elementos de governança centralizada e descentralizada.

A abordagem híbrida foi comum entre as empresas que nós estudamos, mas exige uma grande atenção da gestão [Weill and Ross, 2004].

[Esta página foi intencionalmente deixada em branco]

Capítulo 5

Complexidade de um Processo

Conteúdo

5 COMPLEXIDADE DE UM PROCESSO	72
5.1 INTRODUÇÃO	72
5.1.1 ABORDAGEM METODOLÓGICA	72
5.2 METAMODELO	74
5.3 OCL	76
5.4 RESTRIÇÕES OCL	77
5.4.1 MÉTRICAS COMPLEXAS	77
5.4.1.1 TAMANHO DO PROCESSO/MODELO.....	78
5.4.1.2 COMPLEXIDADE DO PROCESSO/MODELO.....	79
5.4.1.3 ESTRUTURA DO PROCESSO/MODELO	84
5.4.1.4 MODULARIZAÇÃO DO PROCESSO/MODELO.....	86
5.4.1.5 MÉTRICAS NOVAS PROPOSTAS	88
5.5 USE	94

Este capítulo formaliza um conjunto de indicadores quantitativos que permitem exprimir a complexidade de um processo, com recurso à linguagem OCL, sobre um metamodelo BPMN.

5 Complexidade de um Processo

5.1 Introdução

Recordemos que esta dissertação visa contribuir para a avaliação do impacto que o perfil de governança de TI tem na complexidade dos seus processos de TI. Nos capítulos precedentes vimos como se pode caracterizar a governança de TI e determinar o seu perfil. Neste capítulo vamos ver como se pode determinar a complexidade dos processos de serviços de TI.

Desenhar e melhorar processos é um aspecto fundamental para o negócio se manter competitivo no mercado actual. As organizações têm sido forçadas a melhorar os seus processos de negócio porque, os clientes estão a exigir melhores produtos e serviços. Quando uma organização adopta uma filosofia de gestão de processos, a melhoria do processo pode acontecer. Para alcançar uma gestão do processo eficaz, é necessário analisar a complexidade dos processos.

Jorge Cardoso, [Cardoso, 2006], define a complexidade do processo como “o grau em que um processo é difícil de analisar, de compreender ou de explicar. A complexidade do processo pode ser caracterizada pelo número e dificuldade das transições, das decisões condicionais e paralelas, da existência de ciclos, do fluxo de dados, dos papéis, das actividades, dos tipos de estruturas de dados e de outras características do processo.” A alta complexidade num processo tem vários inconvenientes indesejáveis, que pode resultar na má compreensão, em erros, em defeitos e em excepções que levam os processos à necessidade de mais tempo para desenvolver, testar e manter. Por exemplo, em Engenharia de Software, verificou-se que os módulos do programa, com altos índices de complexidade, têm uma maior frequência de falhas.

A complexidade dos processos está intuitivamente ligada aos efeitos da legibilidade, do esforço, da testabilidade, da confiabilidade e da facilidade de manutenção. Assim, analisar a complexidade em todas as etapas do processo de desenho e desenvolvimento, ajuda a evitar os inconvenientes associados à alta complexidade dos processos. Portanto, é importante desenvolver métricas para identificar a complexidade dos processos. Actualmente, as organizações não têm adoptado métricas de complexidade, como parte dos seus projetos de gestão do processo. Como resultado, pode acontecer que simples processos tenham sido desenhados de uma forma complexa [Cardoso, 2006].

5.1.1 Abordagem Metodológica

Para determinarmos a complexidade dos modelos de processos de serviços de TI iremos usar a técnica de *Metamodel Driven Measurement* (M2DM), em que as métricas de complexidade serão formalizadas com a linguagem *OCL* (*Object Constraint Language*) [OCL, 2003] sobre um metamodelo da linguagem de modelação de processos BPMN (*Business Process Modelling Language*). As vantagens da utilização desta técnica são (i) a clarificação do domínio alvo (devida à

utilização de um metamodelo), (ii) a precisão conseguida pela utilização da linguagem OCL, (iii) a disponibilidade de instrumentos automáticos de colheita das métricas que usam como entrada a especificação das métricas a recolher, como o USE (*UML-Based Specification Environment*) aqui usado e, finalmente, (iv) facilita a sempre necessária replicação de experiências.

A técnica do M2DM foi inicialmente proposta em [Brito e Abreu, 2001] no metamodelo GOODLY [Brito e Abreu, et al., 1997]. A técnica M2DM foi utilizada em vários contextos, tais como o de métricas para o desenho de sistemas orientados a objectos especificadas sobre o metamodelo UML 1.x [Baroni, et al., 2002, Baroni, et al., 2003], métricas sobre esquemas de base de dados objecto-relacionais especificadas sobre o *Common Warehouse Metamodel* (CWM) [Baroni, et al., 2004], métricas para componentes de software especificadas sobre o metamodelo UML 2.0 [Goulão, et al., 2004b, Goulão, et al., 2004a] ou sobre *CORBA Component Model* (CCM) [Goulão, et al., 2005b, Goulão, et al., 2005a].

Para usar a técnica de M2DM na avaliação da complexidade dos processos, é utilizado um metamodelo BPMN, expresso através de diagramas de meta-classes em UML e robustecido com regras de boa formação expressas em OCL. Este metamodelo foi proposto em [Freitas, 2010] e é baseado no trabalho em curso no *Object Management Group* (OMG) [<http://www.bpmn.org/>]. A linguagem BPMN permite expressar as actividades dos processos de gestão de serviços TI numa notação gráfica e é recorrentemente escolhida para esse efeito no mundo empresarial.

O trabalho subjacente a este capítulo foi desenvolvido no grupo QUASAR, no âmbito de uma equipa que incluía outros dois mestrados. A figura 5.9 ajuda a clarificar as fases que constituem todo o processo necessário para exprimir a complexidade de um processo, e a forma como o trabalho subjacente a esta dissertação está interligada com o das dissertações de mestrado de Jorge Freitas (JF / Meta-modelação) [Freitas, 2010] e Jose Costa (JC / Instanciação) [Costa, 2010].

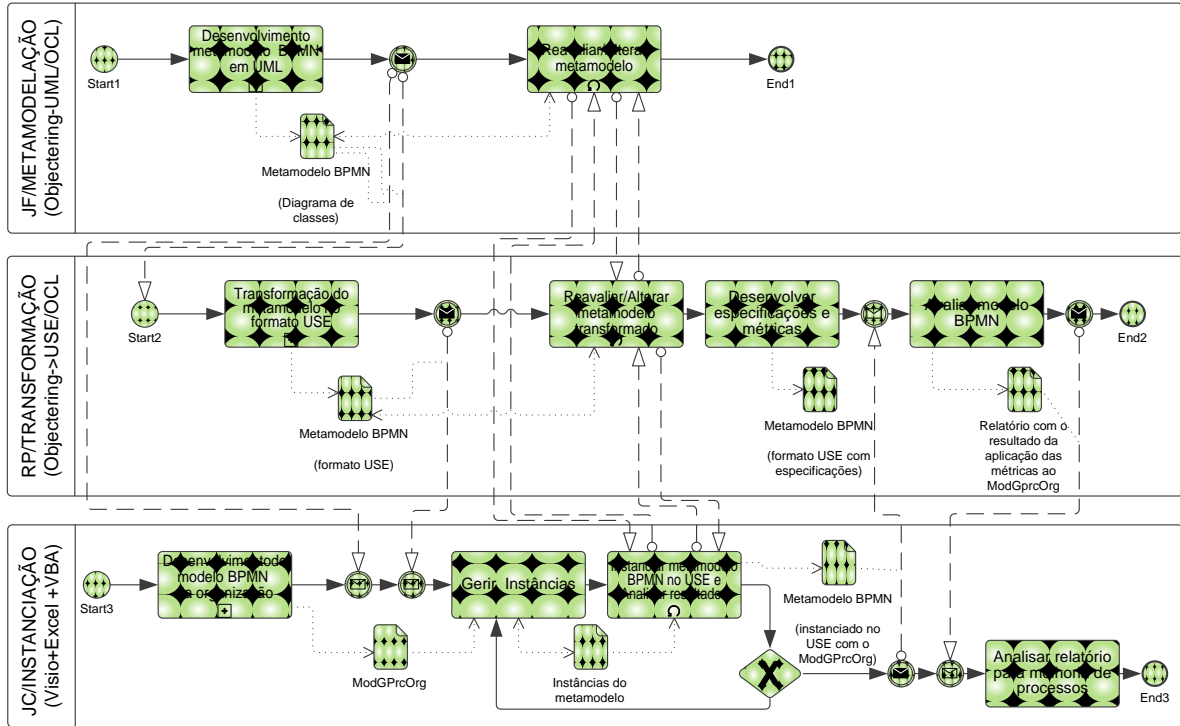


Figura 5.9: Fase de metamodelação, transformação e instanciação

5.2 Metamodelo

Para calcular a complexidade dos processos, é utilizado um metamodelo BPMN (*Business Process Modeling Notation*) e a linguagem OCL (*Object Constraint Language*) para a formalização de métricas adequadas usando a técnica de *Metamodel Driven Measurement* (M2DM), proposta em [Brito e Abreu, 2001].

Para realizarmos uma avaliação quantitativa temos que ser capazes de expressar algumas variáveis descritivas em mais do que uma escala ordinal no nosso domínio de discurso (DoD). Para tal, temos de concordar com aquilo em que estamos a falar, ou seja, precisamos de uma representação bem compreendida do nosso DoD. Aqui é onde as ontologias aparecem. Ontologia é a ciência dos tipos e das estruturas dos objectos, das propriedades e das relações em todos os domínios da realidade [Smith, 2003]. No entanto, em Ciência da Computação, nós usamos a palavra "ontologia" de uma maneira mais leve, para designar uma representação abstracta de conceitos relevantes e das suas relações num determinado domínio. Se esse domínio é o da modelação, então, em vez de falarmos de "ontologia", dizemos simplesmente "metamodelo". Por outras palavras, um metamodelo é uma ontologia da modelação com uma dada linguagem. Existem várias maneiras de definir formalmente uma ontologia e línguas genéricas, bem como domínios específicos, por exemplo, OWL (*Web Ontology Language*) de [McGuinness, et al., 2004]. No âmbito da meta-modelação, a linguagem de especificação mais utilizada é a UML (*Unified Modeling Language*), utilizando diagramas de meta classes enriquecidos com restrições expressas com a *Object Constraint Language* (OCL) [Cranefield, et al., 1999]. Tal

5.3 OCL

OCL (*Object Constraint Language*) é uma linguagem fortemente tipificada. Os seus tipos encontram-se organizados numa hierarquia, conforme representado na figura 5.11, extraída da sua especificação [OCL, 2003].

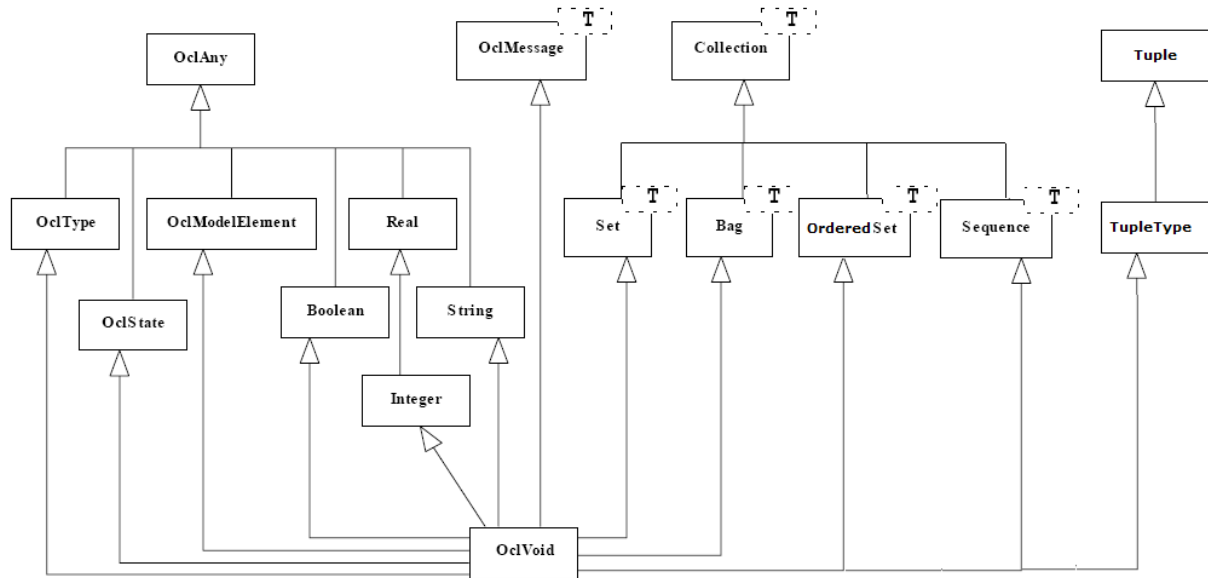


Figura 5.11: Tipos definidos na biblioteca padrão OCL

Os principais tipos de OCL estão divididos em: tipos primitivos, tipos colecção e tuplos. Os tipos primitivos são: *Boolean*, *Integer*, *Real* e *String*. Os tipos colecção são: *Collection* (colecção de elementos do mesmo tipo, duplicados ou não, ordenados ou não); *Set* (conjunto de elementos do mesmo tipo, não-ordenados, sem duplicados, conjunto matemático); *Bag* (como *Set*, mas permite duplicados; multi-conjunto); *OrderedSet* (como *Set*, mas os seus elementos possuem um índice interno de ordenação; conjunto ordenado); *Sequence* (como *Bag*, mas os seus elementos possuem índice interno de ordenação). Um esquema é apresentado na tabela 5.29.

	Sem repetição	Com repetição
Sem ordenação	Set	Bag
Com ordenação	OrderedSet	Sequence

Tabela 5.29: Tipos colecção OCL

Já os tuplos (tipo *TupleType*) são composições de valores, cujos tipos desses valores podem ser diferentes, como por exemplo: *Tuple {name='John', age=10} 1*.

As expressões em OCL não possuem efeitos colaterais: apenas retornam um valor, não alterando nenhuma informação do modelo. A sua avaliação é instantânea – os estados dos objectos não podem ser alterados durante essa avaliação. Além disso, utilizando expressões OCL, é possível a navegação entre as classes de um modelo UML, a partir de uma classe-base, que fornece o contexto da expressão.

Para aceder a um atributo ou operação, navega-se pelas associações utilizando o operador “.”. Para aceder a propriedades de uma colecção, que pode ser definida pelas instâncias de uma classe, ou pelas instâncias ligadas a uma dada instância de uma classe, utiliza-se o operador “->”.

5.4 Restrições OCL

Uma vez que já identificámos a necessidade de um metamodelo, agora vamos voltar a nossa atenção para a definição das métricas. As métricas devem ser formalmente definidas para evitar subjectividade na sua interpretação e na implementação de instrumentos de colheita das mesmas.

Nesta dissertação, vamos utilizar a técnica do *Metamodel Driven Measurement* (M2DM), onde as métricas são definidas como expressões OCL no metamodelo adoptado. Para além da formalidade concedida pela utilização desta linguagem de restrições, as expressões OCL podem ser avaliadas mediante a instanciação do metamodelo utilizando o USE (*UML-Based Specification Environment*).

A técnica do M2DM foi inicialmente proposta em [Brito e Abreu, 2001] no metamodelo GOODLY [Brito e Abreu, et al., 1997]. A técnica M2DM foi utilizada em vários contextos diferentes, tais como os de definição da concepção de métricas sobre sistemas orientados a objectos baseadas no metamodelo UML 1.x [Baroni, et al., 2002, Baroni, et al., 2003], métricas que expressam o esquema de base de dados de objectos relacionais baseadas no *Common Warehouse Metamodel* (CWM) [Baroni, et al., 2004], avaliar componentes de reutilização sobre o metamodelo UML 2.0 [Goulão, et al., 2004b, Goulão, et al., 2004a] ou avaliar componentes de composição utilizando os componentes do metamodelo CORBA (CCM) [Goulão, et al., 2005b, Goulão, et al., 2005a].

Existem várias métricas que serão apresentadas na secção seguinte. É importante mencionar que, no desenvolvimento destas métricas, tentámos criá-las com as seguintes características [Wendy, et al., 1994], [Wendy, et al., 1995]:

- **Simples** – a definição e a utilização da métrica é simples;
- **Objectiva** - diferentes pessoas irão obter valores idênticos das mesmas entidades que estão a ser avaliadas, permitindo a coerência e evitando o enviesamento interpretativo individual;
- **Facilmente colecionáveis** - o custo e o esforço para obter a métrica é razoável;
- **Robusta** - a métrica é insensível a mudanças irrelevantes; permite comparações úteis;
- **Válida** - a métrica mede o que é suposto medir; promove credibilidade da medida.

5.4.1 Métricas Complexas

Os modelos de processo servem como uma base para a comunicação entre os *stakeholders*. Para cumprir este objectivo, esses modelos devem ser fáceis de compreender e fáceis de manter. Por esta razão, é importante avaliarmos a sua complexidade. Embora existam centenas de medidas de complexidade de software que têm sido descritas e publicadas por vários investigadores ao longo das últimas décadas, medir a complexidade dos modelos do processo é uma área de investigação bastante nova, com apenas um pequeno número de contribuições. Baseamo-nos no artigo de [Ghani, 2008] – “*Complexity Metrics for Measuring the Understandability and Maintainability of Business Process*

Models using Goal-Question-Metric (GQM)” que garante que cada métrica tem uma finalidade, e que nenhuma métrica é definida sem um propósito. O autor analisou e resumiu a adaptação de métricas complexas de Engenharia de Software que foram aplicadas em modelos de processos em 5 categorias: tamanho, complexidade, estrutura, compreensibilidade e modularização.

Nesta dissertação, vamos utilizar 4 das 5 categorias descritas por [Ghani, 2008], pois as métricas sobre compreensibilidade ainda estão a ser estudadas (ver tabela 5.30).

Categoria	Métrica de complexidade	Descrição
Tamanho do Processo/Modelo	Nº de actividades	Número total de actividades do modelo
	CFC - <i>Control Flow Complexity</i>	Número de fluxos de controlo no modelo
Complexidade do Processo/Modelo	HPC – <i>Halstead-based process Complexity</i>	As métricas de Halstead medem o comprimento, o volume e a dificuldade do processo
	CNC - <i>Coefficient of Network Complexity</i>	Complexidade de um grafo
Estrutura do Processo/Modelo	<i>Nesting depth</i>	Profundidade de aninhamento na definição de subprocessos
Modularização do Processo/Modelo	HKM - <i>Henry and Kafura Metric (Fan-in / Fan out)</i>	Entradas e saídas do processo (interacções do processo com outros processos)

Tabela 5.30: Métricas complexas

Para além das métricas apresentadas na tabela 5.30, enriquecemos o metamodelo BPMN com mais algumas métricas novas (ver secção 5.4.1.5), para ilustrar a viabilidade da nossa abordagem quantitativa, no contexto da avaliação da complexidade dos processos de TI.

5.4.1.1 Tamanho do Processo/Modelo

No Anexo B secção b) mostramos uma tabela que apresenta as métricas para calcular o tamanho de um **Processo**. Um Processo pode ser composto por *gateways*, actividades, eventos e artefactos que estão ligados entre si através de conectores. Os Processos podem estar em *pools* ou *lanes* diferentes. Um processo também pode dar origem a um ou mais sub-processos.

Ainda no Anexo B secção b) mostramos uma tabela que enuncia as métricas para calcular o tamanho de um **Modelo**. Um Modelo pode ser composto por um ou mais Processos. Estas métricas foram desenvolvidas a partir das métricas calculadas para os Processos. O código referente às métricas para calcular o tamanho de um Processo/Modelo, também se encontra no Anexo B.

5.4.1.2 Complexidade do Processo/Modelo

▪ CFC – *Control Flow Complexity*

O “*cyclomatic number*”, introduzido por Tom McCabe [McCabe, 1976] é o mais utilizado na medição de programas de software. A “*Cyclomatic Complexity Metric*” (CCM) indica se o programa é fácil de compreender e de modificar. Esta métrica é calculada a partir do controlo de fluxo do grafo e mede o número de caminhos linearmente independentes:

$$CCM = \text{número de ramos} - \text{número de nós} + \text{número de pontos de entrada e de saída}$$

Jorge Cardoso [Cardoso, 2006] sugeriu uma medida de complexidade para BPM, que é uma generalização do “*cyclomatic number*” de McCabe. A métrica CFC (*Control Flow Complexity*) foi baseada na análise de *XOR-splits*, *OR-splits* e *AND-splits*. A ideia principal por detrás da métrica CFC definida por Cardoso, é o número de estados mentais que têm de ser considerado aquando da criação e desenvolvimento de um processo. Matematicamente, a métrica de complexidade do fluxo de controlo é aditiva, assim, é muito fácil calcular a complexidade de um processo, adicionando, simplesmente, a CFC de todas as construções de ramificação (*split*)³². A complexidade do fluxo de controlo é calculada como se segue, onde “*P*” é um processo e “*a*” uma actividade.

$$\begin{aligned} CFC(P) = & \sum_{a \in P, a \text{ is a xor-split}} CFC_{XOR}(a) \\ & + \sum_{a \in P, a \text{ is a or-split}} CFC_{OR}(a) + \sum_{a \in P, a \text{ is a and-split}} CFC_{AND}(a) \end{aligned}$$

No nosso metamodelo BPMN os *XOR-splits*³³ são *Data-Based*³⁴ *Exclusive Gateways* e *Event-Based*³⁵ *Exclusive Gateways*, os *OR-splits* são *Data-Based Inclusive Gateways*³⁶ e as *AND-splits* são *Parallel Gateways*³⁷.

Usando a abordagem M2DM, esta métrica foi definida como *complexityProcessCFC()* / *complexityModelCFC()*. A sua definição formal encontra-se no Anexo B. De seguida, exemplificamos

³² Com base no comportamento dos fluxos, a gateway é classificada em quatro tipos. Cada tipo de *gateway* tem dois tipos de representação, uma para uma ramificação baseada numa decisão e outra correspondente à fusão (convergência de ramos do processo).

³³ XOR - *Exclusive Gateways* seleccionam um *Sequence Flow* de diversos *Sequence Flows*. As *Exclusive Gateways* são classificadas em *Data-Based* e *Event-Based*.

³⁴ Dependendo se a expressão condicional proposta por uma saída de *Sequence Flow* for verdadeira ou falsa, é seleccionado um *Sequence Flow* a partir de diversos *Sequence Flows*. Entre Gateways, este tipo é o mais utilizado.

³⁵ Dependendo dos eventos que vêm de fora, é seleccionado um *Sequence Flow* a partir de diversos *Sequence Flows*.

³⁶ OR - *Inclusive Gateways* seleccionam alguns *Sequence Flows* que verificam a condição (a expressão condicional é verdadeira) dos diversos *Sequence Flows*. Pelo menos um *Sequence Flow* deve ser seleccionado.

³⁷ AND - *Parallel Gateways* seleccionam as acções dos *Sequence Flows* de uma vez.

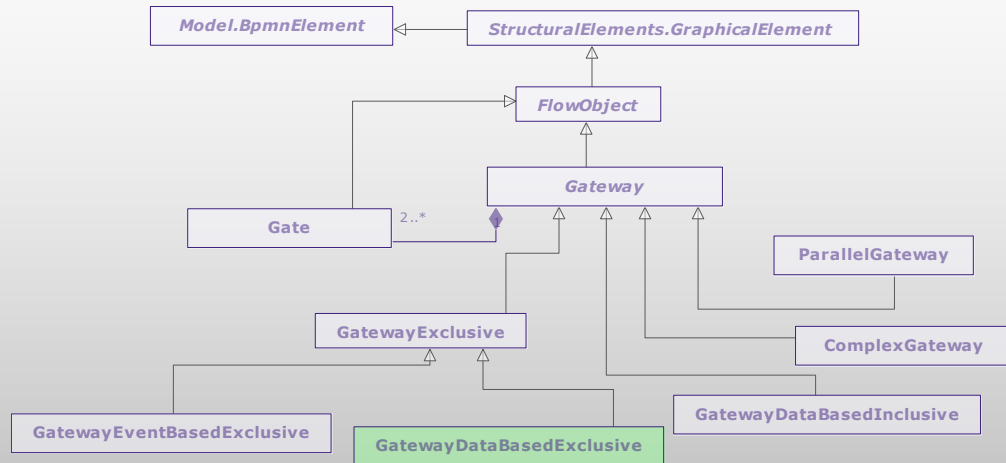
Uma *Complex Gateway* é usada quando os outros três tipos de gateways não são adequados para o processo. A fim de continuar o processamento, pelo menos um *Sequence Flow* deve ser seleccionado.

o cálculo de uma das métricas, *CFC_XOR_DataBased()*, através de um trecho do metamodelo relevante para o cálculo da métrica e uma breve explicação de cada uma das funções que ela usa.

➡ Métrica *CFC_XOR_DataBased()* de um processo

Context: class Model_Process <
SupportingElements_SupportingElement, Orchestration_OrchestrationElement

-- Calcula os XORs Data-based de um processo



```

cFC_XOR_DataBased(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_GatewayDataBasedExclusive)).
    oclAsType(FlowObjects_GatewayDataBasedExclusive)->asSet() ->
    select(isSplit()) -> collect(numberOutputGates()) -> sum()
  
```

1. Selecciona todos os objectos do tipo "GatewayDataBasedExclusive"
2. Retira todos os duplicados (asSet())
3. Selecciona apenas as gates que são Split
isSplit() : Boolean = numberOutputGates() > 1
 -- Número de Splits - conta o número de Gates de Output de uma Gateway
numberOutputGates() : Integer = self.gate-> select(type=#GateType_Output)->size()
4. Retorna a soma dos elementos.

Figura 5.12 – Métrica *CFC_XOR_DataBased()* de um processo

Quanto maior for o valor de *CFC XOR-splits*, *CFC OR-splits* e *CFC AND-splits*, mais complexo é um processo, uma vez que o criador do processo tem de lidar com todos os estados entre os fluxo de controlos construídos (*splits*) e as suas actividades e transições associadas. A análise CFC visa avaliar a complexidade sem a execução directa dos processos.

As vantagens da métrica CFC é que esta pode ser usada como uma métrica de manutenção e de qualidade, dá a complexidade relativa do desenho do processo e é fácil de aplicar. As desvantagens da métrica CFC incluem a incapacidade de medir a complexidade dos dados pois, apenas a complexidade do fluxo de controlo é medida. Além disso, o mesmo peso é colocado sobre *loops* aninhados e não-

aninhados. No entanto, estruturas condicionais, profundamente aninhadas, são mais difíceis de compreender que as estruturas não-aninhadas.

▪ **HPC – *Halstead-based process Complexity***

A medida de Halstead [Halstead, 1987] é outra medida de complexidade de software. Esta métrica foi desenvolvida para determinar uma medida quantitativa da complexidade baseada na compreensão de um programa como uma função dos seus operandos e operadores. As métricas de Halstead compreendem um conjunto de medidas primitivas ($n1$, $n2$, $N1$ e $N2$) que podem ser obtidas a partir do código fonte:

- $n1$ = número de operadores únicos (*if, while, repeat, switch, etc*);
- $n2$ = número de operandos únicos (variáveis ou constantes);
- $N1$ = número total de ocorrências dos operadores;
- $N2$ = número total de ocorrências dos operandos.

Jorge Cardoso [Cardoso, 2006] sugeriu mapear elementos do processo de negócio para o conjunto de medidas primitivas proposto por Halstead:

- $n1$ = número de actividades, *joins, splits* únicos (elementos de fluxos de controlo);
- $n2$ = número de variáveis de dados únicas (*DataObjects*);
- $N1$ = número total de ocorrências dos elementos de fluxos de controlo;
- $N2$ = número total de ocorrências de variáveis de dados.

Com estas medidas primitivas, Cardoso, introduziu a noção de medidas “*Halstead-based process Complexity*” (HPC) para estimar o comprimento, o volume e a dificuldade dos processos. Estas medidas são calculadas da seguinte forma:

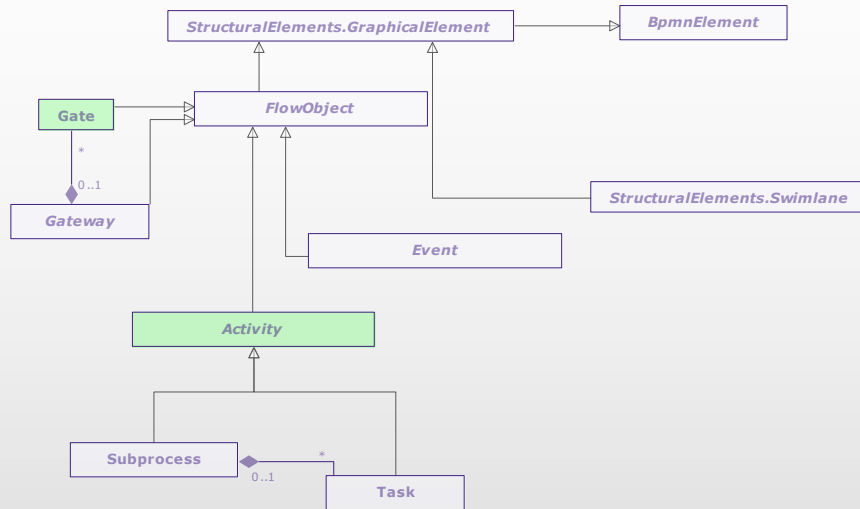
- Comprimento do processo: $C = n1 * \log_2(n1) + n2 * \log_2(n2)$
- Volume do processo: $V = (N1+N2) * \log_2(n1+n2)$
- Dificuldade do processo: $D = \frac{n1}{2} * \frac{N2}{n2}$

Usando a abordagem M2DM, esta métrica foi definida como *length()*, *volume()* e *difficulty()* / *lengthModel()*, *volumeModel()* e *difficultyModel()*. A sua definição formal encontra-se no Anexo B. De seguida, exemplificamos o cálculo de uma das métricas, *length()*, através de um trecho do metamodelo relevante para o cálculo da métrica e uma breve explicação de cada uma das funções que ela usa.

➡ Métrica *length()* de um processo

Context: class Model_Process <
SupportingElements_SupportingElement, Orchestration_OrchestrationElement

-- Calcula o comprimento de um processo: $N = n1 * \log_2(n1) + n2 * \log_2(n2)$



length(): Real = $n1() * \logarithm(2, n1()) + n2() * \logarithm(2, n2())$

1. Calcula o *length()* através da fórmula: $n1 * \log_2(n1) + n2 * \log_2(n2)$, onde:

n1 é o número de actividades, splits e joins únicos. Calculado através das seguintes funções:

n1(): Integer = existActivities() + existSplits() + existJoins()

existActivities(): Integer = if totalNumberProcessActivities() >= 1 then 1 else 0 endif

- verifica se existe, pelo menos uma actividade (subprocess ou task) no processo, através da função

totalNumberProcessActivities(): Integer = self.bpmnElements()->select(oclIsKindOf(FlowObjects_Activity))->size()
Selecciona todos os objectos do tipo "Activity" e retorna a soma dos elementos.

existSplits(): Integer = if totalNumberProcessOutputGates() > 1 then 1 else 0 endif

- verifica se existe, pelo menos um *split* no processo, através da função

totalNumberProcessInputGates(): Integer = self.bpmnElements()->select(oclIsKindOf(FlowObjects_Gateway)).
oclAsType(FlowObjects_Gateway)->asSet->collect(numberOutputGates())->sum()

existJoins(): Integer = if totalNumberProcessInputGates() > 1 then 1 else 0 endif

- verifica se existe, pelo menos um *join* no processo, através da função

totalNumberProcessInputGates(): Integer = self.bpmnElements()->select(oclIsKindOf(FlowObjects_Gateway)).
oclAsType(FlowObjects_Gateway)->asSet->collect(numberInputGates())->sum()

n2 é o número de variáveis de dados únicas (dataObject). Calculado através das seguintes funções:

n2(): Integer = existDataObjects() + existTextAnnotations()

existDataObjects(): Integer = if totalNumberProcessDataObjects() >= 1 then 1 else 0 endif

- verifica se existe, pelo menos um *dataObject* no processo, através da função

totalNumberProcessDataObjects(): Integer = self.bpmnElements()->select(oclIsKindOf(Artifacts_DataObject))->size()

existTextAnnotations(): Integer = if totalNumberProcessTextAnnotations() > 1 then 1 else 0 endif

- verifica se existe, pelo menos um *textAnnotation* no processo, através da função

totalNumberProcessTextAnnotations(): Integer = self.bpmnElements()->select(oclIsKindOf(Artifacts_TextAnnotation))->size()

Figura 5.13 – Métrica *length()* de um processo

Segundo o autor, as medidas HPC tem várias vantagens: não exigem uma análise aprofundada das estruturas do processo; podem prever a taxa de erros e o esforço de manutenção; são fáceis de calcular; e aplicáveis para a maioria das linguagens de modelação de processos.

▪ **CNC - *Coefficient of Network Complexity***

No campo da análise de redes, o “*Coefficient of Network Complexity*” referido nesta dissertação como CNC, é uma métrica amplamente utilizada para avaliar a complexidade da rede [Pascoe, 1966]. O CNC oferece uma métrica bastante simples para a complexidade de um grafo. Esta métrica pode ser facilmente calculada como o número de arcos divididos pelo número de nós. A sua definição original é:

$$CNC = \frac{\text{número de arcos}}{\text{número de nós}}$$

Notou-se que duas redes, com igual número de arcos e nós, mas com diferentes topologias (e, portanto, diferentes níveis de complexidade), não podem ser diferenciadas usando simplesmente esta métrica [Elmaghraby, et al., 1980]. Apesar disto, foram propostas algumas variações desta métrica [Davies, 1974] [Kaimann, 1974] [Kaimann, 1975], e são ainda frequentemente utilizadas na literatura, na sua versão original [Vanhoucke, et al., 2008].

No contexto do nosso metamodelo BPMN, o número de arcos é dividido pelo número de actividades, eventos, *joins* e *splits*:

$$CNC = \frac{\text{número de arcos}}{\text{número de actividades, eventos, joins e splits}}$$

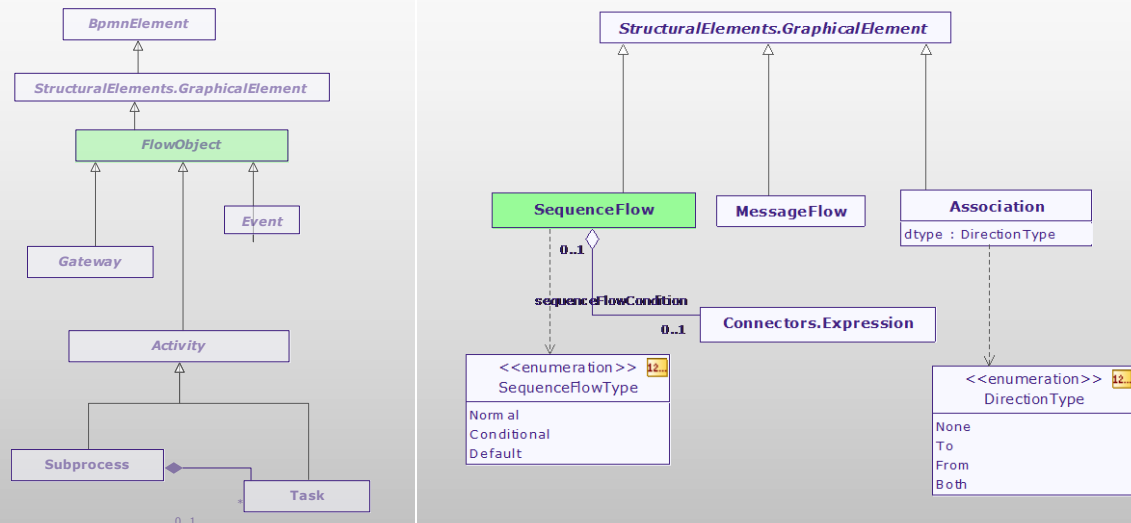
Usando a abordagem M2DM, esta métrica foi definida como *complexityProcessCNC()* / *complexityModelCNC()*. A sua definição formal encontra-se no Anexo B. De seguida, exemplificamos o cálculo da métrica, *complexityProcessCNC()*, através de um trecho do metamodelo relevante para o cálculo da métrica e uma breve explicação de cada uma das funções que ela usa.

➡ Métrica CNC de um processo

Context: class Model_Process <

SupportingElements_SupportingElement, Orchestration_OrchestrationElement

-- Métrica CNC (Coefficient of Network Complexity) = total de Connectors / total de FlowObjects (actividades, eventos e gateways) de um processo



complexityProcessCNC () : Real = totalNumberProcessSequenceFlows() / totalNumberProcessFlowObjects()

Calcula-se o *complexityProcessCNC ()* através da divisão das seguintes funções:

1. **totalNumberProcessSequenceFlows()** que conta o número total de SequenceFlows de um processo
`totalNumberProcessSequenceFlows(): Integer = self.bpmnElements()->
select(oclsKindOf(Connectors_SequenceFlow)) ->size()`
 Selecciona todos os objectos do tipo "SequenceFlow" e retorna a sua soma.
2. **totalNumberProcessFlowObjects()** que conta o número total de FlowObjects (gateways, activities e events) de um processo
`totalNumberProcessFlowObjects(): Integer = self.bpmnElements()->
select(oclsKindOf(FlowObjects_FlowObject)) ->size()`

Figura 5.14– Métrica CNC de um processo

A teoria dos grafos fornece um conjunto rico de métricas de grafos ou de medidas de grafos que podem ser adaptadas para o cálculo da complexidade do processo do grafo.

5.4.1.3 Estrutura do Processo/Modelo

▪ **Nestthing depth**

Volker afirma que os modelos que contêm maior profundidade de aninhamento (*nesting depth*) possuem maior complexidade [Volker, 2007]. A profundidade de aninhamento de um elemento depende do número de decisões no fluxo de controlo que são necessárias para alcançar esse elemento. Alguns modelos podem ter numerosos fluxos de controlo para percorrer até obterem uma decisão, a fim de chegar a um resultado definitivo. Um modelo com mais aninhados pode ser mais complexo e

mais difícil de compreender do que quase todos os modelos lineares, mas o CFC para ambos os modelos poderia ser o mesmo. Por este motivo, Volker também sugeriu a utilização da métrica *nesting depth* para obter o valor *nesting depth* e adicionar o valor ao CFC, a fim de, medir a complexidade do BMP.

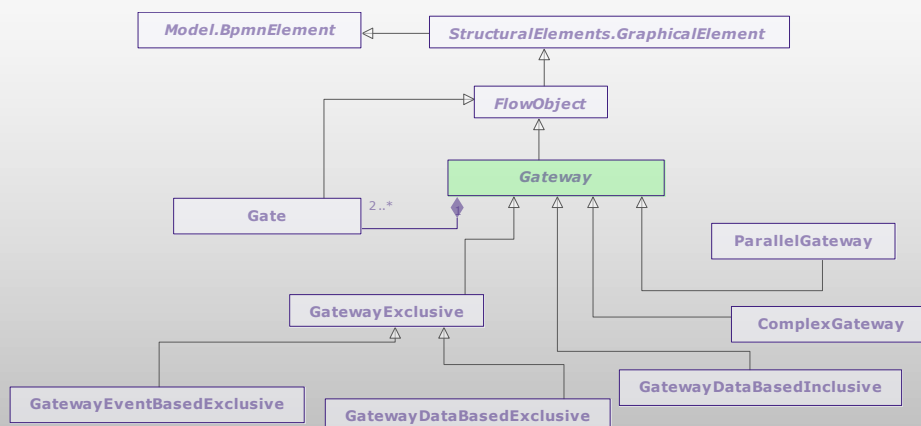
$$\text{Complexidade BPM} = \text{Nesting depth} + \text{CFC}$$

Usando a abordagem M2DM, esta métrica foi definida como *complexidadeBPM* (*origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject*). A sua definição formal encontra-se no Anexo B. De seguida, exemplificamos o cálculo de uma das métricas, *nesting depth()*, através de um trecho do metamodelo relevante para o cálculo da métrica e uma breve explicação de cada uma das funções que ela usa.

➡ Métrica *nesting depth* de um processo

Context: class Model_Process <
SupportingElements_SupportingElement, Orchestration_OrchestrationElement

-- A Métrica *Nesting depth* deve percorrer um dado caminho, com um início e um destino, e contar o número de gateways



```
nestingDepth(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =  
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gateway)) ->  
    collect(oclAsType(FlowObjects_Gateway)) -> size
```

1. Começamos por percorrer todos os caminhos possíveis, desde a origem até ao destino introduzidos, obtendo um conjunto de sub-conjuntos dos vários caminhos possíveis. A função **compute (origin, destination)** está definida na subsecção “Métricas Novas”.
2. A função “flatten” transforma um conjunto de sub-conjuntos num conjunto de objectos individuais.
3. Sobre este conjunto de objectos individuais, seleccionamos todos os objectos do tipo “Gateway”
4. Retorna-se a soma dos elementos.

Figura 5.15– Métrica *nesting depth* de um processo

Uma das diferenças entre um modelo bem estruturado e um modelo mal estruturado são os *joins* e os *splits*. No modelo bem estruturado, os *splits* e *joins* são contidos totalmente dentro da estrutura de controlo, ao passo que no modelo mal estruturado pode ter um salto para fora do bloco de controlo. Em BPM, não ser bem estruturado significa, informalmente, que existe um desajuste entre os conectores *split* e *join* [Ghani, 2008].

5.4.1.4 Modularização do Processo/Modelo

- **Henry and Kafura Metric (HKM)**

A modularização é suportada por quase todas as linguagens de modelação de processos. Ao dividirmos um modelo em sub-modelos modulares, podemos aumentar a sua compreensibilidade e também obter modelos menores, mais passíveis de serem reutilizados [Ghani, 2008].

Henry e Kafura [Henry e Kafura, 1981] propuseram uma métrica baseada no impacto do fluxo de informação na estrutura do programa. A técnica sugere identificar o número de chamadas ao módulo (ou seja, os fluxos de informação locais que entram: *Fan-In*) e identificar o número de chamadas a partir de um módulo (ou seja, os fluxos de informação locais que saem: *Fan-Out*). O seu valor, para cada componente, é definido como:

$$HKM = size \times (Fan-In \times Fan-Out)^2$$

A medida de dimensão (*size*) pode ser escolhida entre várias alternativas, tais como as *lines of code*, *function points* ou CCM do componente. Henry e Kafura validaram a sua métrica utilizando o sistema UNIX e sugeriram que esta permite identificar potenciais componentes defeituosos, uma vez que eles descobriram que elevados valores de HKM eram frequentemente obtidos para componentes onde um maior número de problemas tinha sido reportado.

Esta métrica pode ser usada da mesma forma para analisar BPMs. Nesta dissertação baseamo-nos no “número total de Actividades de um Processo/Modelo” para calcular a dimensão. No contexto do nosso metamodelo BPMN, esta métrica é definida como:

$$HKM = \text{número total de actividades} \times (\text{número de } StartEvents \times \text{número de } EndEvents)^2$$

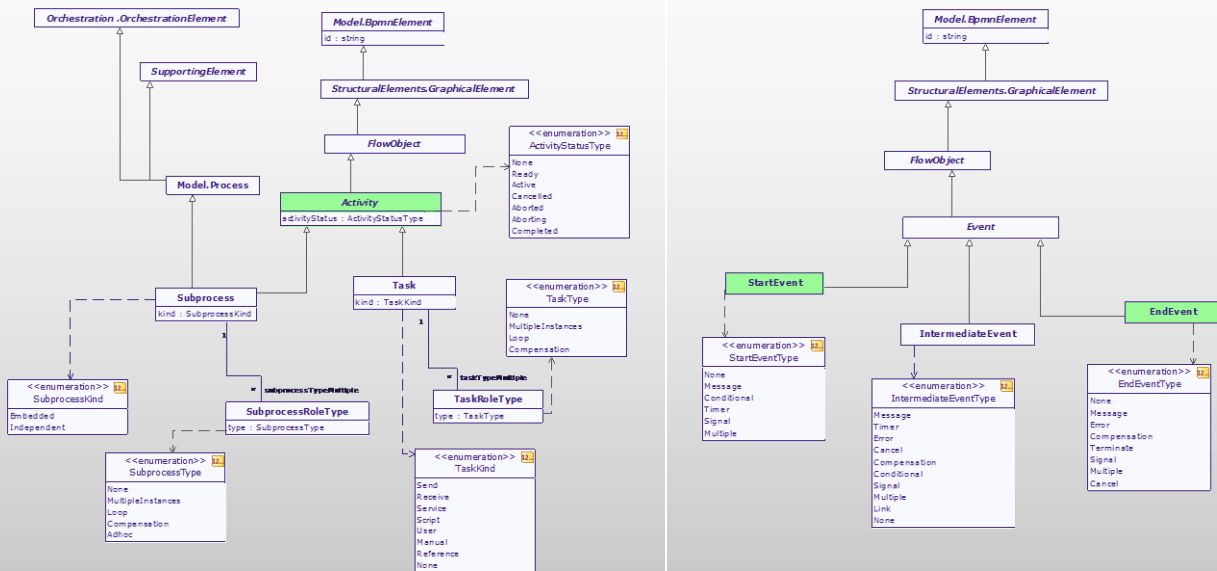
O *fan-in* e *fan-out* pode ser mapeado directamente com os *inputs* e *outputs* das actividades. As actividades são invocadas quando os seus *inputs* (*fan-in*) estão disponíveis e são agendadas para execução. Quando uma actividade completa a sua execução, os seus dados de *output* são transferidos para as ligações das actividades através das suas transições.

Usando a abordagem M2DM, esta métrica foi definida como *modularizationProcessHKM()* / *modularizationModelHKM()*. A sua definição formal encontra-se no Anexo B. De seguida, exemplificamos o cálculo da métrica, *modularizationProcessHKM()*, através de um trecho do metamodelo relevante para o cálculo da métrica e uma breve explicação de cada uma das funções que ela usa.

➡ Métrica HKM de um processo

Context: class Model_Process <
SupportingElements_SupportingElement, Orchestration_OrchestrationElement

-- Métrica HKM = número total de actividades (tasks e subprocessos) x (nº de startevents (fan-in) x
nº de endevents (fan-out))^2



modularizationProcessHKM(): Real = totalNumberProcessActivities() *
power((totalNumberProcessStartEvents() * totalNumberProcessEndEvents()), 2)

Calcula-se a *modularizationProcessHKM()* através das seguintes funções:

1. **totalNumberProcessActivities ()** conta o número total de Activities (Subprocesses+Tasks) de um processo
totalNumberProcessActivities ():Integer = self.bpmnElements()->
select(oclIsKindOf(FlowObjects_Activity))->size()
Selecciona todos os objectos do tipo "Activity" e retorna a sua soma.
2. **totalNumberProcessStartEvents ()** que conta o número total de StartEvents de um processo
totalNumberProcessStartEvents(): Integer = self.bpmnElements()->
select(oclIsKindOf(FlowObjects_StartEvent)) ->size()
Selecciona todos os objectos do tipo "StartEvent" e retorna a sua soma.
3. **totalNumberProcessEndEvents ()** que conta o número total de EndEvents de um processo
totalNumberProcessEndEvents(): Integer = self.bpmnElements()->
select(oclIsKindOf(FlowObjects_EndEvent)) ->size()
Selecciona todos os objectos do tipo "EndEvent" e retorna a sua soma.

Figura 5.16– Métrica HKM de um processo

O elevado valor para *fan-in* é causado pelo módulo ter sido chamado e por ser usado por outro módulo muitas vezes, e um elevado valor de *fan-out* é causado pelo módulo usar ou importar outros módulos muitas vezes [Volker, 2007].

A desvantagem da métrica é que pode dar valores de complexidade igual a zero, caso uma actividade não tenha interacções externas. Isto normalmente só acontece com as actividades finais de um processo.

5.4.1.5 Métricas novas propostas

Nesta subsecção apresentamos um conjunto de novas métricas que propomos com o intuito de enriquecer o metamodelo BPMN e ilustrar a viabilidade da nossa abordagem quantitativa, no contexto da avaliação da complexidade dos processos de TI.

Começamos por definir uma função que percorre todos os caminhos possíveis de um processo e retorna um conjunto de sequências de todos os elementos (*FlowObjects*, *Connectors* e *Artifacts*). Depois, programamos uma função muito semelhante, mas que retorna apenas um conjunto de sequências dos *FlowObjects* (actividades, eventos e *gateways*) desses caminhos (figura 5.17):

compute(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject)

Sendo um modelo de um processo, na realidade um grafo orientado, isto é uma implementação do algoritmo recursivo de atravessamento de grafos em profundidade primeiro (*depth first*).

Context: class Model_Process

-- Função que calcula todos os caminhos possíveis, dando a origem e o destino

compute(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :

```
Set(Sequence(FlowObjects_FlowObject)) =
  visit( origin,
        destination,
        oclEmpty(Set(FlowObjects_FlowObject))->including(origin),
        oclEmpty(Sequence(FlowObjects_FlowObject))->append(origin),
        oclEmpty(Set(Sequence(FlowObjects_FlowObject))))
```

```
visit ( x: FlowObjects_FlowObject,
       destination: FlowObjects_FlowObject,
       visited: Set(FlowObjects_FlowObject),
       path: Sequence(FlowObjects_FlowObject),
       paths: Set(Sequence(FlowObjects_FlowObject))) : Set(Sequence(FlowObjects_FlowObject)) =
  if x=destination then paths->including(path)
  else
    if x.successors()->isEmpty() then paths
    else
      x.successors()->iterate(elem: FlowObjects_FlowObject;
                           acc: Set(Sequence(FlowObjects_FlowObject)) = paths |
                           if visited->excludes(elem) then
                             visit(elem, destination, visited->including(elem), path->append(elem), acc)
                           else
                             acc
                           endif)
      endif
    endif
```

Figura 5.17 - Função que calcula todos os caminhos possíveis dando uma origem e um destino

A partir da função *compute* elaborámos funções que nos permitem saber informação sobre o maior e menor caminho de um processo (figura 5.18) e métricas que, dando uma origem e um destino, nos permitem calcular (figura 5.19 e figura 5.20):

- o número de caminhos possíveis;
- o número total de elementos ou só de *FlowObjects* existentes nos vários caminhos separadamente;
- o número total de elementos ou só de *FlowObjects* existentes num único processo (permite-nos fazer uma análise de cobertura de todo o processo).

Context: class Model_Process

-- Função que retorna o maior caminho. Em caso de empate retorna o primeiro dos maiores caminhos

biggerPath (origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):

```
Sequence(FlowObjects_FlowObject) =
  compute (origin, destination) -> iterate(elem: Sequence(FlowObjects_FlowObject);
    acc: Sequence(FlowObjects_FlowObject) = oclEmpty(Sequence(FlowObjects_FlowObject))
    if (elem->size() > acc->size()) then
      elem
    else
      acc
    endif)
```

-- Função que conta o número de objectos do maior caminho de um processo

countObjectsBiggerPath(origin: FlowObjects_FlowObject,
destination: FlowObjects_FlowObject): Integer =
 biggerPath(origin, destination) -> size()

-- Função que retorna o menor caminho. Em caso de empate retorna o último dos menores caminhos

smallerPath (origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):

```
Sequence(FlowObjects_FlowObject) =
  compute (origin, destination) -> iterate(elem: Sequence(FlowObjects_FlowObject);
    acc: Sequence(FlowObjects_FlowObject) = biggerPath(origin, destination)|
    if (elem->size() > acc->size()) then
      acc
    else
      elem
    endif)
```

-- Função que conta o número de objectos do menor caminho de um processo

countObjectsSmallerPath(origin: FlowObjects_FlowObject,
destination: FlowObjects_FlowObject): Integer =
 smallerPath(origin, destination) -> size()

Figura 5.18 – Funções que nos permitem saber informação sobre o maior e menor caminho de um processo

Context: class Model_Process

-- Função que conta o número de caminhos possíveis de um processo

```
numberPaths(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject): Integer =  
    compute(origin, destination) -> size()
```

-- Função que mostra todos os FlowObjects dos vários caminhos possíveis de um processo (análise de cobertura)

```
allElements(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):  
    Set(FlowObjects_FlowObject) =  
        compute(origin, destination) -> flatten
```

-- Função que conta todos os FlowObjects dos vários caminhos possíveis de um processo

```
countAllElements(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject): Integer =  
    compute(origin, destination) -> flatten -> size()
```

-- Função que conta o número de FlowObjects em todos os caminhos possíveis de um processo

```
countObjectsPaths(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):  
    Bag(Integer) = compute(origin, destination) ->  
        collect(oclIsKindOf(FlowObjects_FlowObject)).count(true)
```

Figura 5.19 – Funções que nos permitem saber informação sobre o número de caminhos possíveis e os *FlowObjects* que estão contidos nesses caminhos

Para além das funções e métricas atrás referidas também podemos calcular, dando uma origem e um destino, o número de:

- Actividades (*Tasks* e *Sub-processos*) – figura 5.20;
 - Gateways – figura 5.21;
 - Gates (*Input* e *Output*) – figura 5.21;
 - Eventos (*Start*, *Intermediate* e *End*) – figura 5.22
- que um determinado caminho contém.

Context: class Model_Process

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de actividades

```
countActivities(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Activity))->
  collect(oclAsType(FlowObjects_Activity))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as actividades contidas nesse caminho

```
seeActivities(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
  Set(FlowObjects_Activity) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Activity))->
    collect(oclAsType(FlowObjects_Activity)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de tasks

```
countTasks(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Task))->
  collect(oclAsType(FlowObjects_Task))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as tasks contidas nesse caminho

```
seeTasks(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Set(FlowObjects_Task) =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Task))->
  collect(oclAsType(FlowObjects_Task)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de subprocessos

```
countSubprocesses(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_SubProcess))->
  collect(oclAsType(FlowObjects_SubProcess))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os subprocessos contidos nesse caminho

```
seeSubprocesses(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
  Set(FlowObjects_SubProcess) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_SubProcess))->
    collect(oclAsType(FlowObjects_SubProcess)) -> asSet()
```

Figura 5.20 – Funções que nos permitem saber informação sobre as Actividades (*Tasks* e *Sub-processos*)

Context: class Model_Process

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de gateways

```
countGateways(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gateway))->
    collect(oclAsType(FlowObjects_Gateway))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as gateways contidas nesse caminho

```
seeGateways(origin:FlowObjects_FlowObject, destination:FlowObjects_FlowObject): Set(FlowObjects_Gateway)=
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gateway))->
    collect(oclAsType(FlowObjects_Gateway)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de gates

```
countGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as gates contidas nesse caminho

```
seeGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Set(FlowObjects_Gate) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de gates de input

```
countInputGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate))->select(type=#Input)->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as gates de input contidas nesse caminho

```
seeInputGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Set(FlowObjects_Gate) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate)) ->select(type=#Input) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de gates de output

```
countOutputGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate))->select(type=#Output)->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as gates de output contidas nesse caminho

```
seeOutputGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):Set(FlowObjects_Gate)=
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate)) ->select(type=#Output) -> asSet()
```

Figura 5.22 - Funções que nos permitem saber informação sobre as *Gateways* e *Gates (Input e Output)*

Context: class Model_Process

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de eventos

```
countEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Event)) ->
  collect(oclAsType(FlowObjects_Event)) -> size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os eventos contidos nesse caminho

```
seeEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):Set(FlowObjects_Event)=
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Event)) ->
  collect(oclAsType(FlowObjects_Event)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de start events

```
countStartEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_StartEvent)) ->
  collect(oclAsType(FlowObjects_StartEvent)) -> size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os start events contidos nesse caminho

```
seeStartEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
  Set(FlowObjects_StartEvent) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_StartEvent)) ->
    collect(oclAsType(FlowObjects_StartEvent)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de intermediate events

```
countIntermediateEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_IntermediateEvent)) ->
  collect(oclAsType(FlowObjects_IntermediateEvent)) -> size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os intermediate events contidos nesse caminho

```
seeIntermediateEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
  Set(FlowObjects_IntermediateEvent) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_IntermediateEvent)) ->
    collect(oclAsType(FlowObjects_IntermediateEvent)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de end events

```
countEndEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_EndEvent)) ->
  collect(oclAsType(FlowObjects_EndEvent)) -> size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os end events contidos nesse caminho

```
seeEndEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
  Set(FlowObjects_EndEvent) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_EndEvent)) ->
    collect(oclAsType(FlowObjects_EndEvent)) -> asSet()
```

Figura 5.23 - Funções que nos permitem saber informação sobre os Eventos (*Start, Intermediate e End Events*)

Todas as funções atrás ilustradas recebem como parâmetro uma origem e um destino. Foram definidas exactamente as mesmas funções recebendo apenas como parâmetro a origem. Isto porque, em exemplos com grandes quantidades de dados, torna-se confuso ver onde começa e termina um processo. Assim passamos como argumento só a origem e obtemos na mesma os valores das métricas atrás definidas.

5.5 USE

Por fim, para compilar o modelo utilizamos o *USE (UML-based Specification Environment)* [<http://www.db.informatik.uni-bremen.de/projects/USE/>] [Gogolla, Bohling and Richters, 2003], [Richters and Gogolla, 2003], [Bauerdick, Gogolla and Gutsche, 2004], [Gogolla, Bohling and Richters, 2005] que é uma ferramenta de validação de modelos de classes UML anotados com OCL, desenvolvida na Universidade de Bremen que permite avaliar o valor de expressões em OCL sobre um modelo (no nosso caso um metamodelo) instanciado com objectos (no nosso caso os gerados no âmbito da dissertação [Costa, 2010]).

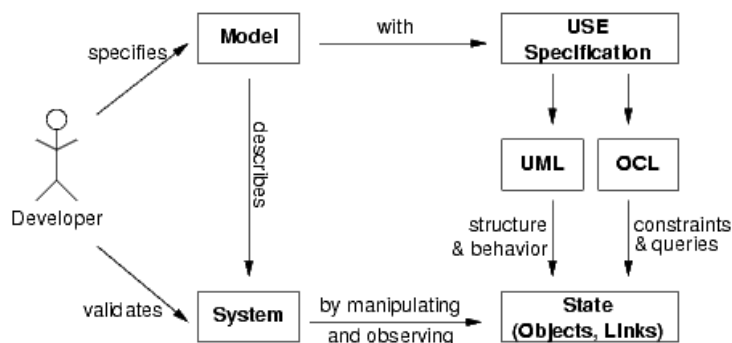


Figura 5.24 – Abordagem USE (<http://www.db.informatik.uni-bremen.de/projects/USE/>)

Um inconveniente do uso da ferramenta é a necessidade de que os dados de entrada – diagrama de classes e suas restrições, bem como a correspondente instanciação (criação de objectos, definição do seu estado e ligações entre os objectos) – estejam num formato textual próprio do USE, não suportando a leitura de um modelo já construído noutro formato, como XMI, por exemplo. Para ultrapassar este obstáculo contamos com o contributo do trabalho desenvolvido no âmbito de [Costa, 2010], no qual, a partir de modelos de processos descritos com a linguagem BPMN na ferramenta *Microsoft Visio* se geraram instâncias do metamodelo BPMN, constituídas por meta-objectos (instâncias das meta-classes) e meta-ligações (instâncias das meta-associações) no formato textual nativo da ferramenta USE. Esta geração foi realizada pelo *Microsoft Excel* e pelo *Visual Basic for Applications*³⁸ (VBA). De seguida, apresentamos um exemplo de criação de três objectos, alteração do seu estado e a criação de uma ligação entre os objectos, no formato textual do USE:

³⁸ VBA - é uma implementação do *Visual Basic* da *Microsoft* incorporado em todos os programas da *suíte Microsoft Office*. Permite aos programadores adequar e expandir as capacidades e funções de aplicativos *desktop Microsoft* para que estes possam ser integrados com outros dados e sistemas. [wikipedia, Visual Basic for Applications]

```

!create Text : Artifacts_TextAnnotation
!create Task : FlowObjects_Task
!create Association : Connectors_Association

```

```

!set Task.name := 'Task'
!set Text.contents := 'Text'
!set Association.dtype:= #DirectionType_From

```

```

!insert (Text, Association) into BpmnElement_Association
!insert (Task, Association) into BpmnElement_Association1

```

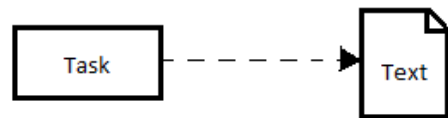


Figura 5.25 – Exemplo textual USE

Usando as instâncias assim criadas, que andam na ordem dos milhares, e para melhor demonstrarmos como uma análise quantitativa pode ser desempenhada, vamos carregar o metamodelo BPMN no USE. A figura 5.25 ilustra a abertura da especificação do metamodelo

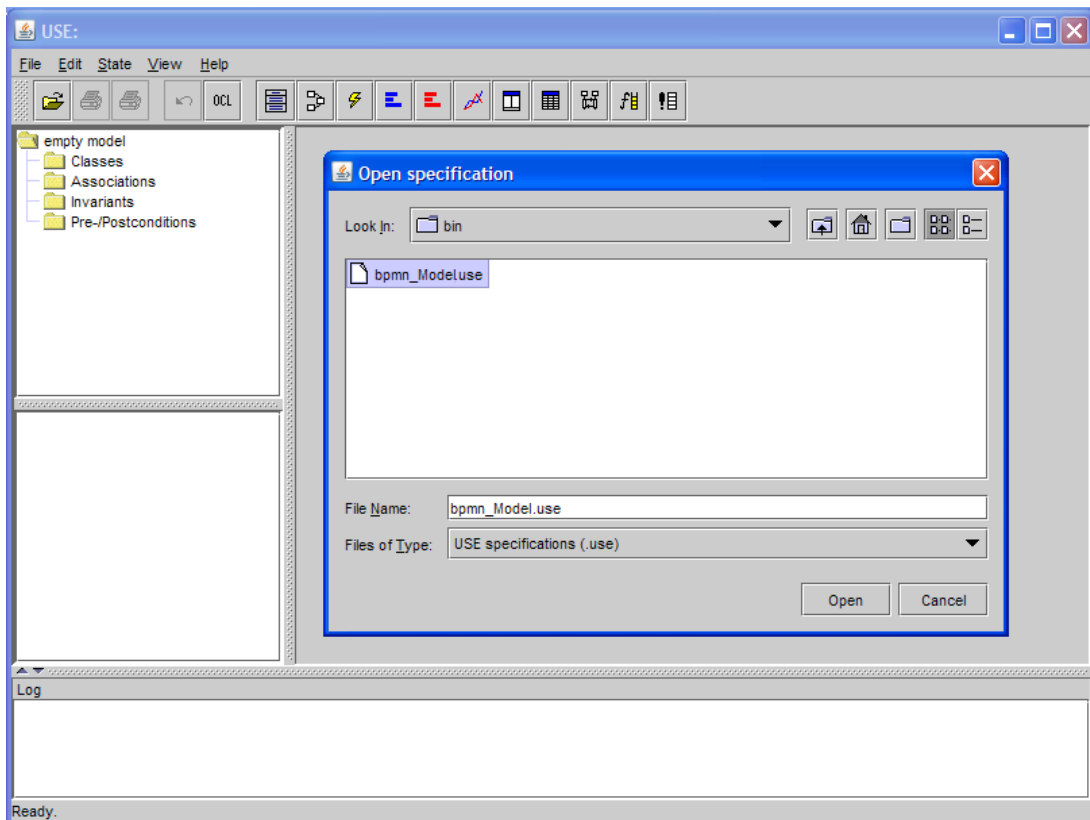


Figura 5.26 – Janela “Open Specification” do USE

Depois do carregamento a especificação é compilada, se esta não tiver erros, o USE exibe no painel de “log” o resultado do processo de compilação e o número de classes, associações, invariantes, operações e pré-/pós condições, conforme ilustrado na figura 5.26. Observe que a nossa especificação contém 337 operações criadas no âmbito desta dissertação.

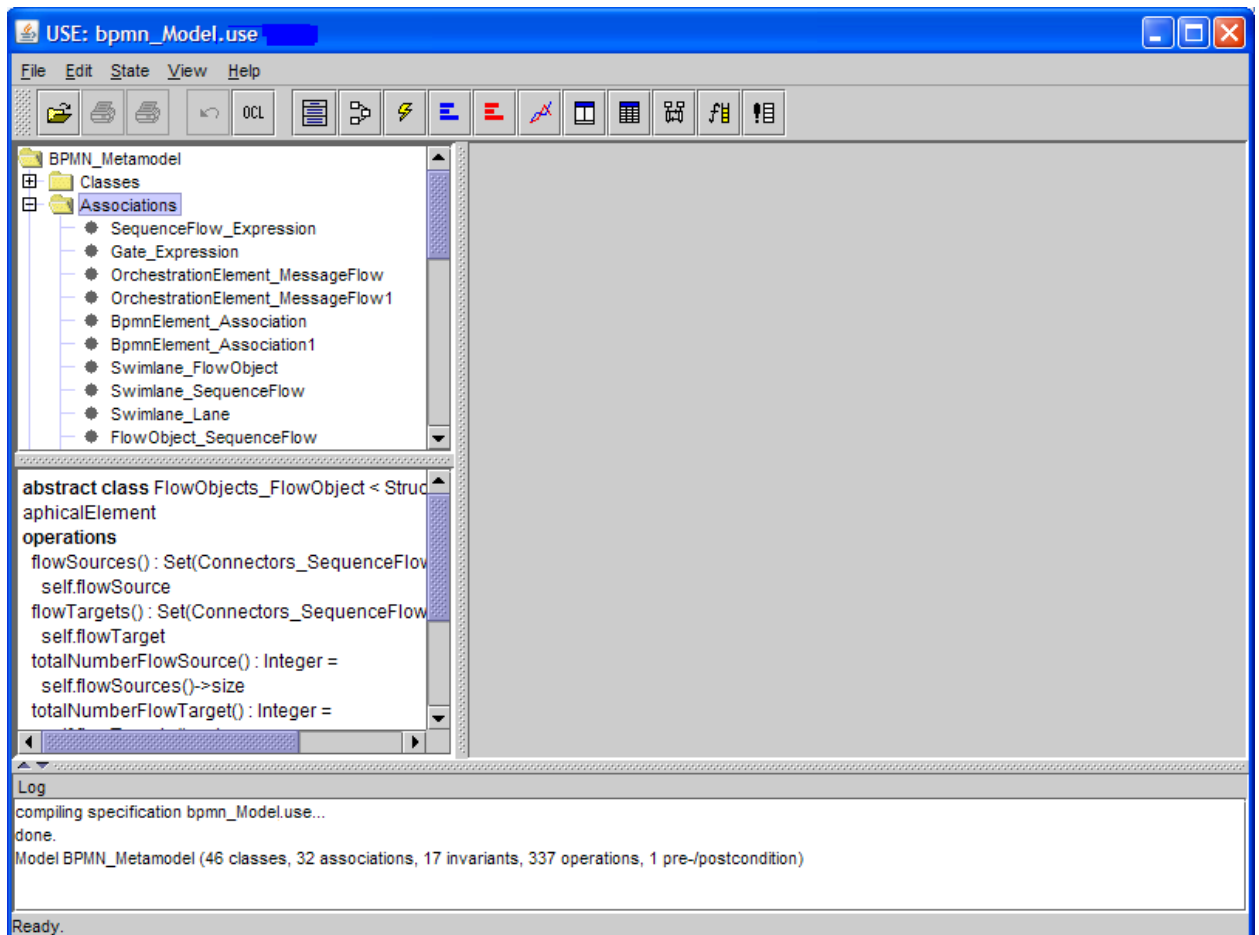
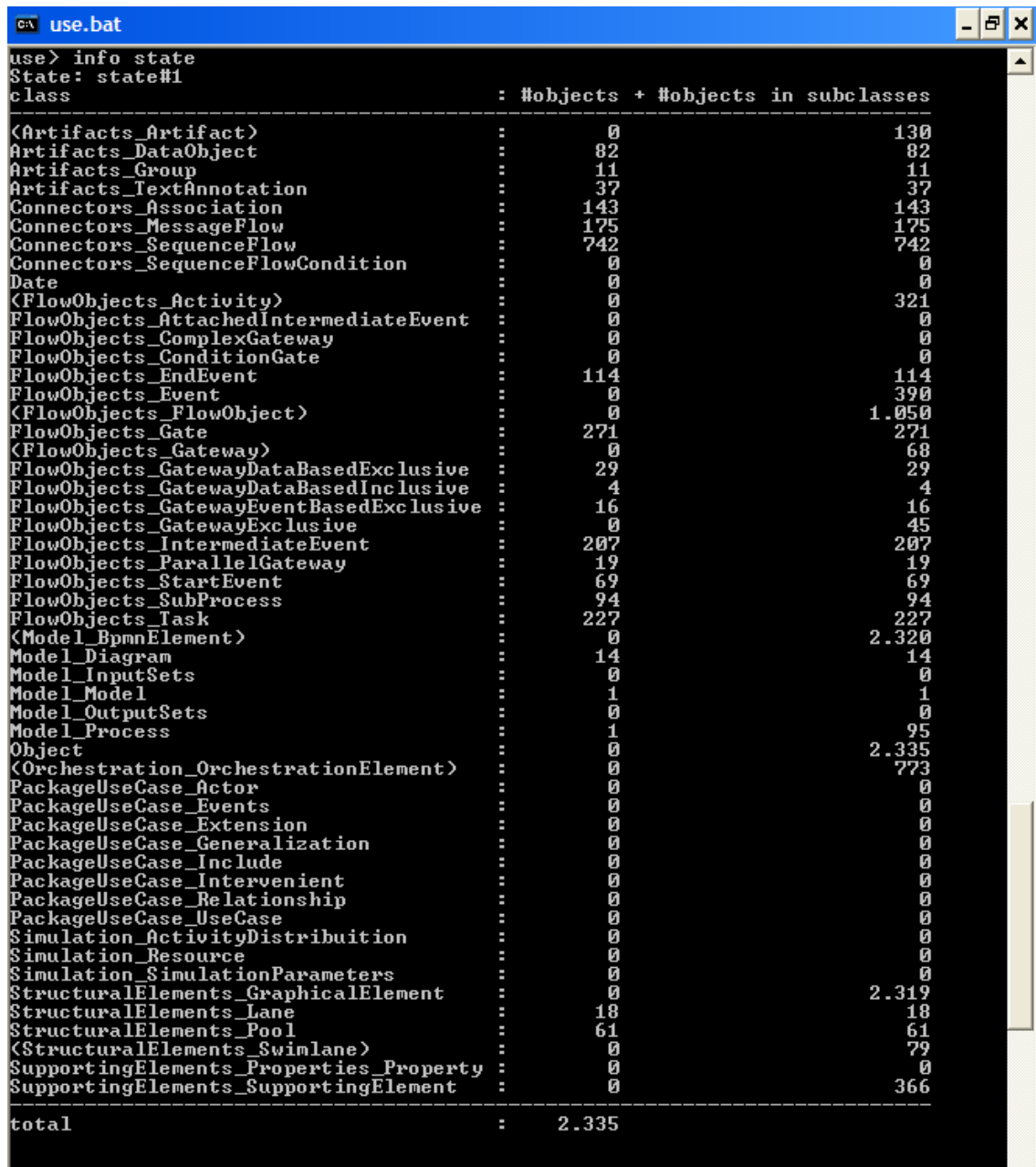


Figura 5.27 - “*bpmn_Model.use*” carregado no USE

O próximo passo consiste no carregamento das instâncias do metamodelo. As figuras 5.27 e 5.28 mostram que neste exemplo carregámos 2.335 objectos e 3.281 associações.

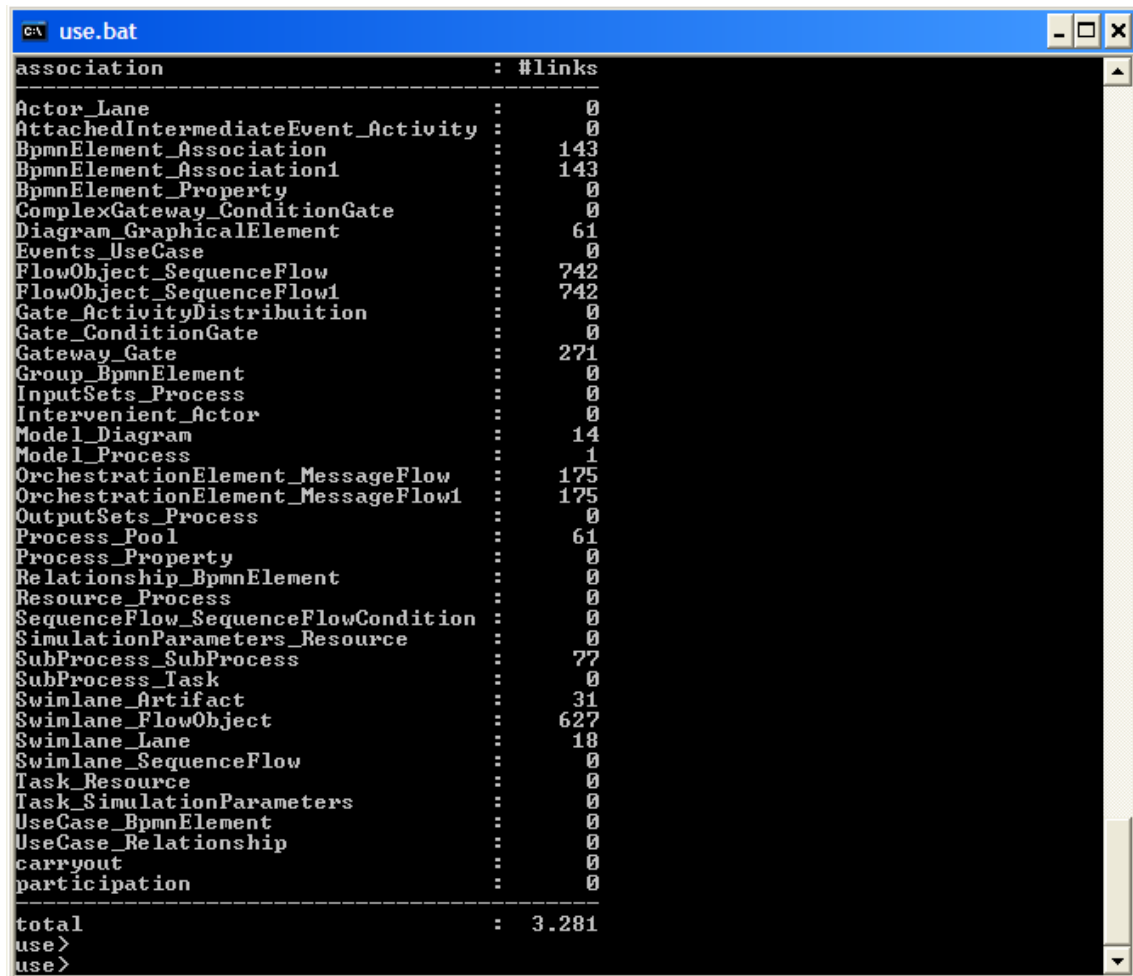


```

C:\ use.bat
use> info state
State: state#1
class                                     : #objects + #objects in subclasses
-----
<Artifacts_Artifact>                     :      0      130
Artifacts_DataObject                     :      82      82
Artifacts_Group                           :      11      11
Artifacts_TextAnnotation                  :      37      37
Connectors_Association                    :     143     143
Connectors_MessageFlow                    :     175     175
Connectors_SequenceFlow                   :     742     742
Connectors_SequenceFlowCondition          :      0      0
Date                                       :      0      0
<FlowObjects_Activity>                   :      0     321
FlowObjects_AttachedIntermediateEvent    :      0      0
FlowObjects_ComplexGateway                :      0      0
FlowObjects_ConditionGate                 :      0      0
FlowObjects_EndEvent                      :     114     114
FlowObjects_Event                         :      0     390
<FlowObjects_FlowObject>                  :      0     1.050
FlowObjects_Gate                          :     271     271
<FlowObjects_Gateway>                    :      0      68
FlowObjects_GatewayDataBasedExclusive    :      29      29
FlowObjects_GatewayDataBasedInclusive    :      4       4
FlowObjects_GatewayEventBasedExclusive   :     16      16
FlowObjects_GatewayExclusive              :      0      45
FlowObjects_IntermediateEvent             :     207     207
FlowObjects_ParallelGateway               :      19      19
FlowObjects_StartEvent                    :      69      69
FlowObjects_SubProcess                    :      94      94
FlowObjects_Task                          :     227     227
<Model_BpmnElement>                      :      0     2.320
Model_Diagram                            :     14      14
Model_InputSets                           :      0      0
Model_Model                               :      1       1
Model_OutputSets                          :      0      0
Model_Process                            :      1      95
Object                                    :      0     2.335
<Orchestration_OrchestrationElement>     :      0      773
PackageUseCase_Actor                     :      0      0
PackageUseCase_Events                     :      0      0
PackageUseCase_Extension                  :      0      0
PackageUseCase_Generalization             :      0      0
PackageUseCase_Include                    :      0      0
PackageUseCase_Intervient                 :      0      0
PackageUseCase_Relationship               :      0      0
PackageUseCase_UseCase                    :      0      0
Simulation_ActivityDistribution           :      0      0
Simulation_Resource                       :      0      0
Simulation_SimulationParameters           :      0      0
StructuralElements_GraphicalElement       :      0     2.319
StructuralElements_Lane                   :     18      18
StructuralElements_Pool                   :     61      61
<StructuralElements_Swimlane>             :      0      79
SupportingElements_Properties_Property    :      0      0
SupportingElements_SupportingElement      :      0     366
-----
total                                     :     2.335

```

Figura 5.28 – Meta-objetos carregados



```

association : #links
-----
Actor_Lane : 0
AttachedIntermediateEvent_Activity : 0
BpmnElement_Association : 143
BpmnElement_Association1 : 143
BpmnElement_Property : 0
ComplexGateway_ConditionGate : 0
Diagram_GraphicalElement : 61
Events_UseCase : 0
FlowObject_SequenceFlow : 742
FlowObject_SequenceFlow1 : 742
Gate_ActivityDistribution : 0
Gate_ConditionGate : 0
Gateway_Gate : 271
Group_BpmnElement : 0
InputSets_Process : 0
Intervient_Actor : 0
Model_Diagram : 14
Model_Process : 1
OrchestrationElement_MessageFlow : 175
OrchestrationElement_MessageFlow1 : 175
OutputSets_Process : 0
Process_Pool : 61
Process_Property : 0
Relationship_BpmnElement : 0
Resource_Process : 0
SequenceFlow_SequenceFlowCondition : 0
SimulationParameters_Resource : 0
SubProcess_SubProcess : 77
SubProcess_Task : 0
Swimlane_Artifact : 31
Swimlane_FlowObject : 627
Swimlane_Lane : 18
Swimlane_SequenceFlow : 0
Task_Resource : 0
Task_SimulationParameters : 0
UseCase_BpmnElement : 0
UseCase_Relationship : 0
carryout : 0
participation : 0
-----
total : 3.281
use>
use>

```

Figura 5.29 - Meta- ligações carregadas

Agora vamos aplicar as métricas definidas anteriormente para obter informações sobre a complexidade dos processos e do modelo instanciado. Para isso usamos a linha de comandos fornecida pela ferramenta USE.

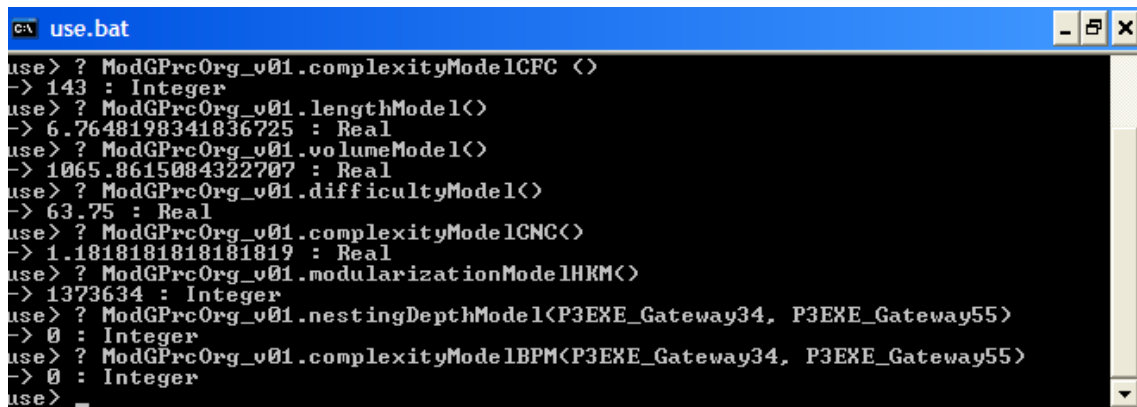

```

C:\ use.bat
use> ?GPrj.totalNumberProcessSwimlanes<
-> 122 : Integer
use> ?GPrj.totalNumberProcessPools<
-> 61 : Integer
use> ?GPrj.totalNumberProcessLanes<
-> 61 : Integer
use> ?GPrj.totalProcessLanes<
-> Set<@APGP_Lane333,@EPGP_Lane333,@EPP_Lane333,@P1INI_Lane30,@P1INI_Lane333,@P1INI_Lane336,@P1INI_Lane92,@P2APLA_Lane30,@P2APLA_Lane333,@P2APLA_Lane335,@P2APLA_Lane30,@P2PLA_Lane333,@P2PLA_Lane335,@P2PLA_Lane336,@P3EXE_Lane,@P4MeC_Lane30,@P4MeC_Lane333,@P5ENC_Lane333> : Set<StructuralElements_Lane>
use> ?GPrj.totalNumberProcessFlowObjects<
-> 627 : Integer
use> ?GPrj.totalNumberProcessActivities<
-> 306 : Integer
use> ?GPrj.totalNumberProcessTasks<
-> 225 : Integer
use> ?GPrj.totalNumberProcessSubProcesses<
-> 81 : Integer
use> ?GPrj.totalNumberProcessEvents<
-> 254 : Integer
use> ?GPrj.totalNumberProcessStartEvents<
-> 67 : Integer
use> ?GPrj.totalNumberProcessEndEvents<
-> 1 : Integer
use> ?GPrj.totalNumberProcessIntermediateEvents<
-> 186 : Integer
use> ?GPrj.totalNumberProcessGateways<
-> 67 : Integer
use> ?GPrj.totalNumberProcessGatewayDataBasedExclusives<
-> 29 : Integer
use> ?GPrj.totalNumberProcessGatewayDataBasedInclusives<
-> 4 : Integer
use> ?GPrj.totalNumberProcessGatewayEventBasedExclusives<
-> 16 : Integer
use> ?GPrj.totalNumberProcessComplexGateways<
-> 0 : Integer
use> ?GPrj.totalNumberProcessParallelGateways<
-> 18 : Integer
use> ?GPrj.totalNumberProcessGates<
-> 267 : Integer
use> ?GPrj.totalNumberProcessInputGates<
-> 110 : Integer
use> ?GPrj.totalNumberProcessOutputGates<
-> 157 : Integer
use> ?GPrj.totalNumberProcessArtifacts<
-> 85 : Integer
use> ?GPrj.totalNumberProcessDataObjects<
-> 75 : Integer
use> ?GPrj.totalNumberProcessTextAnnotations<
-> 10 : Integer
use> ?GPrj.totalProcessTextAnnotations<
-> Set<@P1INI_TextAnnotation,@P2PLA_TextAnnotation,@P3EXE_TextAnnotation4,@P4MeC_TextAnnotation157,@P4MeC_TextAnnotation377,@P4MeC_TextAnnotation57,@P5ENC_TextAnnotation,@P5ENC_TextAnnotation303,@P5ENC_TextAnnotation377,@P5ENC_TextAnnotation4> : Set<Artifacts_TextAnnotation>
use> ?GPrj.totalNumberProcessGroups<
-> 0 : Integer
use> ?GPrj.totalNumberProcessSequenceFlows<
-> 741 : Integer
use> ?GPrj.totalNumberProcessMessageFlows<
-> 97 : Integer
use> ?GPrj.totalNumberProcessAssociations<
-> 0 : Integer
use> ?GPrj.totalNumberProcessConnectors<
-> 838 : Integer

```

Figura 5.30 – Exemplificação de algumas métricas no USE

O resultado apresentado na figura 5.29 é o efeito da aplicação de algumas métricas apresentadas na subsecção “5.4.1 Métricas de Complexidade”. Existem diversas variações para estas métricas complexas como, por exemplo, mostrar o nome dos elementos BPMN, em vez de mostrar o número dos elementos e vice-versa, ou aplicar estas métricas a um processo ou a um modelo. Na figura 5.29 aplicamos as métricas a um Processo, com o nome “GPrj”. Já na figura 5.30 optámos por aplicar as métricas a um Modelo com o nome “ModGPrOrg_v01”.



```
G:\ use.bat
use> ? ModGPrOrg_v01.complexityModelCFC (>
-> 143 : Integer
use> ? ModGPrOrg_v01.lengthModel(>
-> 6.7648198341836725 : Real
use> ? ModGPrOrg_v01.volumeModel(>
-> 1065.8615084322707 : Real
use> ? ModGPrOrg_v01.difficultyModel(>
-> 63.75 : Real
use> ? ModGPrOrg_v01.complexityModelCNC(>
-> 1.1818181818181819 : Real
use> ? ModGPrOrg_v01.modularizationModelHKM(>
-> 1373634 : Integer
use> ? ModGPrOrg_v01.nestingDepthModel(P3EXE_Gateway34, P3EXE_Gateway55)
-> 0 : Integer
use> ? ModGPrOrg_v01.complexityModelBPM(P3EXE_Gateway34, P3EXE_Gateway55)
-> 0 : Integer
use>
```

Figura 5.31 - Exemplificação de algumas métricas complexas no USE

Agora que já temos o metamodelo e as métricas definidas e testadas com muitas instâncias e, conseqüentemente já conseguimos calcular a complexidade de um processo, estamos prontos para analisar casos de estudos e verificar, se existe de alguma forma, alguma relação com o perfil de governança de TI e a complexidade dos processos de uma organização. Esta última análise é detalhada no próximo capítulo “Análise dos Resultados”.

Capítulo 6

Análise dos Resultados

Conteúdo

6 ANÁLISE DOS RESULTADOS	102
6.1 INTRODUÇÃO	102
6.2 ANÁLISE DE CORRELAÇÃO	102
6.2.1 INTRODUÇÃO	102
6.2.2 ANÁLISE DE RESULTADOS	103
6.3 CASO DE ESTUDO	105
6.3.1 CARACTERIZAÇÃO DO ORGANISMO	105
6.3.2 ANÁLISE E RESULTADOS DO QUESTIONÁRIO DE GOVERNANÇA DE TI	106
6.3.3 COMPLEXIDADE DOS PROCESSOS	111
6.3.4 CONCLUSÕES DO CASO DE ESTUDO	114
6.4 IMPACTO DA GOVERNANÇA NOS PROCESSOS	115

Neste capítulo são analisados os resultados obtidos da análise de correlação entre as variáveis das métricas e é apresentado um caso de estudo de um organismo público português em que se relaciona a governança de TI com a complexidade dos processos de TI.

6 Análise dos Resultados

6.1 Introdução

Para analisarmos e avaliarmos a governança de TI (através dos perfis de governança), a complexidade dos processos de TI (através das métricas implementadas) e a relação entre eles, vamos apresentar neste capítulo:

- uma análise de correlação das métricas descritas e implementadas que nos permitem medir a complexidade de um processo; o intuito desta análise é constatar se as métricas seleccionadas permitem ou não avaliar aspectos diferenciados da complexidade dos modelos de processo;
- um caso de estudo de um organismo português, onde foram:
 - distribuídos e recolhidos inquéritos de governança de TI ao director e coordenadores de departamento, e que nos permitiram definir um perfil de governança de TI;
 - recolhidos dados dos processos de Gestão de Incidentes e de Gestão de Problemas, actualmente implementados neste organismo, e que nos permitiram instanciar e medir a complexidade destes processos.

6.2 Análise de Correlação

6.2.1 Introdução

Nesta análise vamos usar o coeficiente de correlação produto-momento (r), ou coeficiente de *Pearson*³⁹. É o coeficiente de correlação mais utilizado, que expressa a magnitude da relação entre duas variáveis, expressas em, pelo menos, uma escala do tipo intervalar. Tal como a generalidade dos coeficientes de correlação, toma valores no intervalo $[-1, 1]$. Uma correlação positiva ocorre quando uma variável aumenta e a outra também aumenta (exemplo: quanto maior for o número de controlos de decisão maior será o número de actividades de um processo) e uma correlação negativa ocorre quando uma variável diminui e a outra aumenta (exemplo: quanto menor for o número de recursos alocados às actividades de um processo, maior será o tempo que essas actividades irão demorar a ser desempenhas).

A investigação correlacional (IC) preocupa-se em determinar a associação que existe entre as variáveis, ou seja, a IC é essencialmente exploratória – tenta identificar o padrão da co-variação existente entre duas ou mais variáveis. Não há manipulação das variáveis. Quando há uma associação de alguma magnitude entre duas variáveis, dizemos que estão correlacionadas. A magnitude e a direcção da relação são descritas por meio de um índice quantitativo chamado coeficiente de

³⁹ Karl Pearson (1857-1936) - foi um grande contribuidor para o desenvolvimento da Estatística como uma disciplina científica séria e independente. Algumas das suas principais contribuições são: teoria da regressão linear e correlação, teste Qui-quadrado de Pearson, coeficiente de correlação e coeficientes de assimetria [Wikipédia, Pearson].

correlação. Relativamente ao valor da magnitude, apresentamos de seguida uma tabela resumo com os critérios interpretativos propostos em [Bento, 2009].

Coefficiente	Interpretação
Entre 0 e 0,20	Correlação praticamente nula
Entre 0,21 e 0,40	Correlação baixa
Entre 0,41 e 0,70	Correlação moderada
Entre 0,71 e 0,90	Correlação forte
Entre 0,91 e 1	Correlação muito forte

Tabela 6.31: Critérios para análise dos índices de correlação

Note-se que a existência de correlação nada diz sobre a natureza da relação causal que porventura exista entre as variáveis. Ao interpretar um coeficiente de correlação deve ter-se presente que um valor elevado não significa que X seja causa de Y, ou Y seja causa de X⁴⁰. Porém, quando existe uma relação causal forte, também se verifica uma correlação forte, pelo que a análise correlacional pode ajudar a descartar possíveis hipóteses teóricas. Não nos devemos contudo esquecer que a lógica de uma relação causal deve provir de teorias externas ao âmbito da Estatística [Malva, 2008].

6.2.2 Análise de Resultados

Para fazermos a análise de correlação das métricas, utilizamos novamente o exemplo oriundo de [Costa, 2010] que foi dividido em 31 subprocessos, tantos quantos os *StartEvents*⁴¹ que lá se podem encontrar.

Para cada um desses subprocessos calculámos, com recurso ao ambiente USE, as métricas indicadas no capítulo precedente. Desta forma obtivemos uma amostra de dimensão considerável⁴², como é requerido pela Estatística⁴³ para garantir uma maior robustez dos resultados.

Utilizamos a ferramenta SPSS *Statistics* 17.0⁴⁴ para fazermos a análise de correlação. No SPSS seleccionamos a opção *Correlate -> Bivariate*. Escolhemos o coeficiente de correlação paramétrico de *Pearson*, pois é o mais poderoso [Bento, 2009] e pode ser utilizado em variáveis quantitativas, mesmo quando os pressupostos da normalidade não são garantidos, desde que a amostra não seja demasiado pequena. Seleccionamos todas as variáveis das métricas. O SPSS apresentou-nos o resultado da análise de correlação como ilustrado na figura 6.31 (é apenas um pequeno extracto, pois a tabela é muito grande):

⁴⁰ Um caso paradigmático disto mesmo é o facto de os dados disponibilizados pelas agências internacionais sobre os vários países apontarem para que exista uma elevada correlação entre o grau de literacia e a esperança de vida média, variáveis estas sobre as quais não se consegue encontrar uma relação de causalidade evidente. A sua elevada correlação advém de ambas estarem relacionadas com o produto per capita de cada país.

⁴¹ Para mais informações sobre os *startEvents* (pontos de partida), por favor ver o capítulo 5 referente ao metamodelo.

⁴² Um dos resultados práticos do Teorema do Limite Central é o tamanho da amostra. O TLC permite afirmar que n suficientemente grande significa, de um modo geral, $n \geq 30$.

<http://www.math.ist.utl.pt/~mjmoraes/NotasApoioPE-SeccaoFolhas.pdf>, pág. 141

⁴³ A Estatística é uma ciência que se dedica à colecta, análise e interpretação de dados. Preocupa-se com os métodos de recolha, organização, resumo, apresentação e interpretação dos dados, assim como tirar conclusões sobre as características das fontes donde estes foram retirados, para melhor compreender as situações [Wikipédia, Estatística].

⁴⁴ Para mais informações sobre o software SPSS *Statistics*, versão 17.0 da IBM, por favor consulte o link: <http://www.spss.com/statistics/changes.htm>

		Nº de caminhos possíveis	Nº de Objectos diferentes em todos os caminhos possíveis
Métrica_-_BPM	Pearson Correlation	,812**	,898**
	Sig. (2-tailed)	,000	,000
	N	31	31
Métrica_HKM	Pearson Correlation	,143	,439*
	Sig. (2-tailed)	,442	,013
	N	31	31

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Figura 6.32 - Programa SPSS *Statistic*

Os resultados só são estatisticamente significativos se o valor de $p < 0.05$ (na figura 6.31 o p está representado como *Sig.(2-tailed)*). Ou seja, se o valor de $p=0.05$, quer dizer que há apenas 5 possibilidades em 100 que a hipótese nula seja verdadeira; Logo, há 95 possibilidades em 100 que a hipótese alternativa seja verdadeira, isto é, 95% de probabilidades que este conjunto de valores utilizado para esta correlação seja estatisticamente significativo. Por exemplo, através da figura 6.31, podemos observar que a métrica HKM e o número de objectos diferentes em todos os caminhos possíveis têm uma correlação moderada de 43,9% e que a amostra tem um número suficiente de dados, capaz de estabelecer uma significância de 98,7% em relação ao grau de correlação calculado.

Relembrando um pouco a Estatística, podemos dizer que se:

- $p > 0.05$, a hipótese nula não é rejeitada, isto é, a correlação não é significativa;
- $p < 0.05$, a hipótese nula é rejeitada, isto é, a correlação é significativa.

No capítulo anterior definimos mais de 30 métricas para calcular a complexidade de um processo. Com o intuito de reduzirmos este número de variáveis e assim o esforço de colheita, se as mesmas forem redundantes, e tendo como base os critérios para análise dos índices de correlação (tabela 6.31), identificámos um conjunto de métricas que estão fracamente correlacionadas entre si (tabela 6.32). Com efeito, se duas métricas tiverem uma correlação muito elevada, isso significa que estão a medir uma mesma faceta da complexidade e por isso podemos dispensar uma delas.

Nas células onde aparece ** é porque a correlação é significativa ao nível 0.01. Nas células onde aparece * é porque a correlação é significativa ao nível 0.05. As variáveis de A a F, tanto na vertical, como na horizontal da tabela 6.33, são as métricas.

ID	Variáveis	A	B	C	D	E	F
A	Nº de Caminhos possíveis		-,099	,002	,054	,143	,539**
B	Nº de End Events			-,047	-,136	,845**	,049
C	Nº de TextAnnotations				,376*	,054	,378*
D	Métrica CNC					-,022	,364*
E	Métrica HKM						,324
F	Métrica HPC - difficulty						

Tabela 6.32: Matriz 6x6 com resultados obtidos através do programa SPSS Statistic

Em termos de conclusão, podemos dizer que é mais que natural que o valor de correlação entre estas métricas seja baixo pois, estão a medir coisas diferentes:

- nº de percursos,
- nº de eventos finais (*endEvents*),
- nº de notas (*textAnnotations*),
- complexidade, através do número de *connectors* e de *flowObjects* (*gateways*, *activities* e *events*),
- interação do processo com outros processos, através do número de *activities*, de *startEvents* e de *endEvents*,
- dificuldade, através do número de *activities*, *gateways*, *dataObjects* e *textAnnotations*,

As métricas “Nº de caminhos possíveis” e “HPC - difficulty” têm valores de correlação elevados entre elas porque dependem fortemente das *gateways*.

As métricas “Nº de endEvents” e “HKM” também têm valores de correlação elevados entre elas pois dependem fortemente dos *endEvents*.

6.3 Caso de Estudo

Este caso de estudo é referente a um organismo público português⁴⁵. A sua apresentação está dividida em três partes: (i) caracterização do organismo em estudo e análise do perfil de governança de TI, (ii) análise da complexidade de dois processos de suporte de serviços de TI, e (iii) conclusões.

6.3.1 Caracterização do Organismo

Este organismo tem como competência dar suporte à Governança no âmbito das Tecnologias da Informação e Comunicação. Compete-lhe garantir a utilização mais eficaz das Tecnologias da Informação e Comunicação, e particularmente da Internet. É constituído por 3 departamentos:

- Departamento responsável pelo suporte Administrativo e Financeiro;
- Departamento responsável pela gestão de qualidade e serviços, desde a prestação dos serviços de apoio localizado e personalizado aos utilizadores, à gestão dos serviços de suporte tecnológico e à disponibilização de serviços. Assim, integra funções de Auditoria e todas as

⁴⁵ Não podemos divulgar o nome deste organismo devido à confidencialidade que nos foi solicitada.

competências organizacionais de Gestão de Utilizadores, Controlo e Operações, *Service Desk*, Gestão de Projectos e Gestão de Serviços;

3. Departamento responsável pela gestão de toda a infraestrutura tecnológica do organismo, ou seja, pelas infraestruturas e serviços de comunicações e *networking*, de sistemas operativos, aplicações e todos os conteúdos Internet. Integra todas as competências organizacionais de Comunicações e Sistemas, Segurança, Aplicações e Sistemas de Apoio à Decisão e *Internet* e Conteúdos, nas diferentes dimensões de apoio e suporte, controlo operacional de infraestruturas e engenharia, investigação e desenvolvimento.

Este organismo é constituído por um director geral, dois coordenadores de departamento, 34 profissionais de TI internos e 12 colaboradores externos. De seguida apresentamos um quadro resumo que caracteriza este organismo público. Esta informação foi recolhida através das perguntas do grupo C do questionário, que se encontra em anexo na secção A.

Caracterização do organismo público em estudo

Sector de Actividade	Governo
Tipo de organização	Prestadora de serviços de TI interna e externamente Consultora integradora
Custos dedicados à TI	100%
Nº de unidades de negócio	5
Nº de Profissionais de TI	34
Total de colaboradores	46 (34 Internos + 12 externos)

Tabela 6.33: Caracterização do organismo público em estudo

6.3.2 Análise e Resultados do Questionário de Governança de TI

Para definirmos o perfil de governança de TI deste organismo distribuámos o inquérito de governança de TI (ver Capítulo 4 e Anexo A) à gestão de topo (director-geral e 2 coordenadores de departamento). Após analisarmos os três questionários, obtivemos os seguintes resultados:

1. Perfil de Governança de TI

Este perfil foi calculado através das respostas à pergunta 1 do grupo A do questionário, que se encontra em anexo na secção A. É referente a quem detém os direitos de decisão e a quem detém os direitos de contribuir para a decisão de cada um dos cinco domínios de decisões (tabela 3.17) e dos seis arquétipos de TI (tabela 3.19) (Secção 3.3 “Como é que as empresas governam a TI: Decisões-chave e Arquétipos de TI para a Governança de TI”).

Com base na figura 3.6 “Os três perfis mais eficazes medidos pelo desempenho da governança de TI”, na tabela 3.19 “Arquétipos de TI” e no valor médio dos 3 inquéritos respondidos, obtivemos o seguinte perfil de governança para o organismo em estudo:







Domínio Arquétipo	Princípios de TI	Arquitectura de TI	Infraestrutura de TI	Necessidades de aplicações de negócio	Investimentos de TI
Monarquia do negócio					
Monarquia de TI					
Feudal					
Federal					
Duopólio					
Anarquia					

Tabela 6.34: Perfil de Governança de TI do organismo em estudo

O perfil a tracejado e a azul é referente ao perfil de governança de TI do organismo público que estamos a estudar. O perfil contínuo a rosa é referente ao perfil 3 *standard* que se encontra na figura 3.6. Este perfil 3 pertence aos três perfis mais eficazes, medidos pelo desempenho da governança de TI referidos na secção 3.5 “Perfis de Governança de TI”. Comparativamente à figura 3.6, podemos concluir que o perfil traçado para este organismo assemelha-se em 80% ao perfil 3, em 20% ao perfil 2 e em 0%, ou seja, em nada ao perfil 1.

Podemos então afirmar que o perfil do organismo em estudo (a tracejado e a azul) está muito próximo do perfil 3, diferindo apenas no domínio de decisão - necessidades de aplicação do negócio, que no nosso estudo não é federal, mas monarquia do negócio.

Como podemos observar através da tabela 6.34, a monarquia do negócio está presente nos cinco domínios de decisão do organismo em estudo. Isto quer dizer que é o director geral do organismo que toma todas as decisões de TI que afectam toda a empresa. Tipicamente, as monarquias do negócio recebem contributos para as decisões-chave de muitas fontes. Geralmente, estes contributos são provenientes: (i) directamente do CIO, (ii) das unidades de negócios, através da equipe de liderança da TI, (iii) do processo de gestão de orçamento de TI, (iv) de acordos de níveis de serviço e da recuperação de custos de TI e (v) de um sistema de acompanhamento que mostra todos os recursos de TI e como eles são utilizados.

Podemos dizer que a monarquia do negócio esta presente em todos os 5 domínios de decisão, porque:

- Princípios de TI** (definição do papel das TIs no negócio da organização (por exemplo: suportar cortes nos custos ou crescimento nas receitas)) - é utilizada, porque o director-geral permite à TI moldar os princípios de negócio, reflectindo as capacidades da TI já existentes ou em desenvolvimento nesses princípios.
- Arquitectura de TI** (opções técnicas relativamente à arquitectura de TI (por exemplo: opção por arquitectura cliente/servidor, *web service*, etc.)) – é utilizada, pois o director tem a percepção de que deve assumir a liderança para garantir que a TI está alinhada com as estratégias de negócio. Na monarquia do negócio, os decisores da TI concentram-se na criação

de uma arquitectura de TI integrada e flexível, orientada pelos princípios de TI do negócio, produzidos pelo director.

- c) **Infraestrutura da TI** (estratégias de implementação/evolução da infraestrutura de TI (por exemplo: redes, hardware, etc.)) – é utilizada, porque a sua pequena dimensão faz com que haja pouca segregação de funções. O director é o único líder de negócio.
- d) **Necessidade de aplicações de negócio** (necessidade de aquisição ou desenvolvimento de novas aplicações para suportar o negócio (*portfolio* aplicacional)) – é utilizada, porque o director sente a necessidade de controlar todas as aquisições ou desenvolvimento de novas aplicações para saber se vão de encontro aos objectivos de negócio. Por sua vez, o perfil 3, em vez de monarquia do negócio, utiliza federal. Este arquétipo permite ter como líderes, os directores ou os gestores de unidades de negócio ou os donos de processos de negócio ou todos.
- e) **Investimentos de TI** (investimento em TI ou em negócios que dependam grandemente das TIs) – é utilizada, porque faz tomadas de decisão *top-down* para implementar uma alteração na estratégia.

Em conclusão, podemos dizer que o perfil de governança de TI do organismo em estudo é muito centralizado, com as monarquias de negócio a assumirem todas as decisões. Esta abordagem mais centralizada é utilizada neste organismo pois a rentabilidade e o controlo de custos é um tema predominante. Este perfil exige que o director seja interessado e bem informado sobre as questões de TI - muitas vezes o resultado de trabalhar em estreita colaboração com os coordenadores de departamento.

Este perfil é sensível, quando estão a ocorrer grandes mudanças (por exemplo: reestruturação, fusões, grandes cortes de custos, crises, etc.) devendo os direitos de decisão ser bem defendidos.

Concluimos, então, que o perfil de governança de TI do organismo em estudo é idêntico ao perfil que obtivemos para as médias empresas (entre 50 e 250 colaboradores). Note-se que (i) este organismo é constituído por 46 colaboradores, sendo praticamente considerado uma média empresa e (ii) no estudo que fizemos no capítulo 4, não considerámos como representativos os resultados, tanto das pequenas empresas, como das médias empresas, pois tínhamos amostras muito reduzidas.

2. Desempenho da governança de TI

Este desempenho foi calculado através das respostas à pergunta 1 e 2 do grupo B do questionário, que se encontra em anexo na secção A. A pergunta 1 avalia a importância de um resultado particular e a pergunta 2 avalia o quão bem a governança de TI contribui para esse resultado.

Conforme podemos ler no capítulo 3, o desempenho de governança de uma empresa é avaliado, calculando a eficácia da governança de TI na prestação de quatro factores ponderados pela sua importância para a empresa (tabela 6.35).

Quando os gestores de topo avaliam o desempenho da governança, primeiro identificam a importância relativa de cada um dos quatro factores na sua empresa; seguidamente classificam o desempenho da empresa em cada factor. Portanto, matematicamente, temos:

Desempenho da governança

$$= \frac{\sum_{n=1}^4 (\text{importância do resultado } \{i. e \text{ Questão } 1\} * \text{influência da governança de TI } \{i. e \text{ Questão } 2\}) * 100}{\sum_{n=1}^4 (5(\text{importância do resultado}))}$$

Nesta dissertação calculamos o desempenho da governança de TI do organismo em estudo para a média dos 3 inquéritos e obtivemos os seguintes resultados:

	--	-	+/-	+	++
Boa rentabilização do investimento em TI		1			2
Uso eficaz da TI para melhorar a utilização dos recursos da organização	1		2		
Uso eficaz da TI para o crescimento do negócio	1			2	
Uso eficaz da TI para conceber maior flexibilidade ao negócio	1			1	1
Total	3	1	2	3	3

Tabela 6.35: Importância dos resultados de Governança de TI

	--	-	+/-	+	++
Boa rentabilização do investimento em TI					3
Uso eficaz da TI para melhorar a utilização dos recursos da organização				1	2
Uso eficaz da TI para o crescimento do negócio					3
Uso eficaz da TI para conceber maior flexibilidade ao negócio					3
Total	0	0	0	1	11

Tabela 6.36: Influência da Governança de TI

Através da fórmula matemática em cima e dos dados alcançados através dos 3 inquéritos (director-geral e os 2 coordenadores de departamento), obtivemos um desempenho de governança de TI de 98,7%. Fizemos as contas separadamente, só para o director geral com um desempenho de 100% e para os dois coordenadores de departamento com um desempenho de 98%. Isto mostra que a visão do director-geral e dos executivos de negócio é muito similar, ou seja, a gestão de topo deste organismo está alinhada.

Pela observação da tabela 6.35 e da tabela 6.36 podemos concluir que os factores de governança de TI que tiveram e têm mais influência são a boa rentabilização do investimento em TI e o uso eficaz da TI tanto para o crescimento do negócio, como para conceber maior flexibilidade ao negócio. O factor com menos importância é o uso eficaz da TI para melhorar a utilização dos recursos (humanos e materiais) do organismo.

3. Mecanismos de Governança de TI

Estes mecanismos foram determinados através das respostas às perguntas 2 e 3 do grupo A do questionário, que se encontra em anexo na secção A. A governança de TI mapeia os tipos de decisões, os arquétipos para a tomada dessas decisões e como essas decisões serão tomadas e monitorizadas. Esta última etapa requer a concepção e implementação de um conjunto coordenado de mecanismos de governança. É com estes mecanismos que os gestores trabalharam diariamente e a governança é promulgada (capítulo 3).

Nas organizações existem, com frequência, uma variedade de mecanismos e de órgãos com maior ou menor intervenção na governança de TI. De seguida, apresentamos a avaliação feita pelo director-geral e pelos dois coordenadores de departamento, relativamente à eficiência desses mecanismos e órgãos para potenciar o valor do negócio das TIs no organismo em estudo:

	--	-	+/-	+	++	NA
Conselho de Administração (constituído maioritariamente pelos directores dos vários sectores da organização)					3	
Comissão de aprovação do orçamento					3	
Conselho de gestão de TI (órgão misto composto por executivos de TI e de negócio)				2	1	
Conselho de administração de TI (órgão exclusivamente composto por executivos de TI)						3
Conselho da arquitectura de TI						3
Equipas de processos com membros de TI				1	2	
Relacionamento de gestores de negócio/gestores de TI				2	1	
Total	0	0	0	5	10	6

Tabela 6.37: Avaliação dos órgãos com intervenção na governança de TI

Como podemos observar pela tabela 6.37 a maioria dos órgãos acima referidos têm importância, à excepção do conselho de administração de TI e do conselho de arquitectura de TI que não são aplicáveis neste organismo. Os órgãos com maior grau de intervenção na governança de TI, do organismo em estudo, são o conselho de administração, a comissão de aprovação do orçamento e as equipas de processos com membros de TI.

Isto acontece porque, como já foi referido, neste organismo, o director é o líder. Assim, tanto o conselho de administração como a comissão de aprovação do orçamento constituem um mecanismo onde as decisões mais importantes são tomadas, o que encoraja uma visão holística de todo o organismo. Já as equipas de processos com membros de TI também têm um elevado grau de intervenção pois têm uma visão do processo usando a TI (e outros activos) efectivamente. O seu comportamento desejável é a gestão do processo, pois desta forma assegura que o comportamento diário é consistente com as políticas e fornece *input* para as decisões.

	--	-	+/-	+	++
Mecanismo de rastreio/avaliação do valor de negócio da TI		1	2		
Portais web e/ou intranets para suporte de governança de TI				2	1
Mecanismo de rastreio/acompanhamento dos projectos de TI e dos recursos gastos em TI				2	1
Existência de Acordos de Nível de Serviço (SLA) de TI		2	1		
Existência de Acordos internos de recuperação de custos de TI	1	1	1		
Total	1	4	4	4	2

Tabela 6.38: Mecanismos de governança de TI

Como podemos observar pela tabela 6.38, para o organismo em estudo, os mecanismos com maior eficiência são os portais *web* e/ou intranets para suporte de governança de TI e, os mecanismo de rastreio/acompanhamento dos projectos de TI e dos recursos gastos em TI. Os mecanismos com menor eficiência na governança de TI são a existência de *SLAs* de TI e de acordos internos de recuperação de custos de TI.

Isto é justificado pela sua missão, pelos seus objectivos de negócio⁴⁶, pela sua natureza e pela sua abordagem centralizada, onde a rentabilidade e o controlo do custo são um tema predominante.

6.3.3 Complexidade dos Processos

O organismo em estudo tem como objectivo ser certificado pelas boas práticas de qualidade nos serviços em Tecnologias da Informação e Comunicações. Para tal, estão a implementar novos processos e a melhorar os existentes. A Gestão de Incidentes e a Gestão de Problemas são dois processos que já estavam implementados (fase *As-Is*) e foram melhorados de acordo com as boas práticas (fase *To-Be*).

Para cada um desses processos calculámos as métricas resultantes do estudo correlacional realizado no início deste capítulo e apresentamos de seguida uma tabela com os valores obtidos:

⁴⁶ A missão e os objectivos de negócio do organismo público em estudo não podem ser descritos devido à confidencialidade dos dados.

	Gestão de Incidentes		Gestão de Problemas	
	<i>As-Is</i>	<i>To-Be</i>	<i>As-Is</i>	<i>To-Be</i>
Nº de Processos	1	1	1	1
Nº de Caminho Possíveis (Start1, End1)	3	10	1	4
Nº de Objectos diferentes em todos os caminhos possíveis	27	44	14	31
Nº de Objectos nos vários caminhos (Start1, End1)	{15,20,21}	{16,16,20,20,24,24,29,29,30,30}	{14}	{25,25,26,26}
Nº de Objectos no caminho maior (Start1, End1)	21	30	14	26
Nº de Objectos no caminho menor (Start1, End1)	15	16	14	25
Nº de Pools	1	1	1	1
Nº de Lanes	4	3	4	3
Nº de Swimlanes	5	4	5	4
Nº de FlowObjects	17	23	13	19
Nº de Activities	12	15	10	13
Nº de SubProcesses	0	0	0	0
Nº de Tasks	12	15	10	13
Nº de Events	2	2	2	2
Nº de Start Events	1	1	1	1
Nº de End Events	1	1	1	1
Nº de Intermediate Events	0	0	0	0
Nº de Gateways	3	6	1	4
Nº de GatewayDataBasedExclusives	3	6	1	4
Nº de GatewayEventBasedExclusives	0	0	0	0
Nº de GatewayDataBasedInclusives	0	0	0	0
Nº de ComplexGateways	0	0	0	0
Nº de ParallelGateways	0	0	0	0
Nº de Gates	10	21	3	12
Nº de InputGates	5	10	1	6
Nº de OutputGates	5	11	2	6
Nº de Artifacts	0	0	0	0
Nº de DataObjects	0	0	0	0
Nº de TextAnnotations	0	0	0	0
Nº de Groups	0	0	0	0
Nº de Connectors	18	27	13	20
Nº de SequenceFlows	18	27	12	20
Nº de MessageFlows	0	0	1	0
Nº de Associations	0	0	0	0
Métrica CFC	4	10	2	4
Métrica HPC - length	4,8	4,8	2	4,8
Métrica HPC - volume	23,8	33,4	11	27,0
Métrica HPC - difficulty	1,5	1,5	1	1,5
Métrica CNC	1,1	1,2	0,9	1,1
Métrica -NestingDepth (Start1, End1)	3	6	1	4
Métrica - BPM	8	17	2	10
Métrica HKM	12	15	10	13

Tabela 6.39: Métricas da Gestão de Incidentes e Gestão de Problemas

Como podemos observar pela tabela 6.39 o número de caminhos possíveis na Fase To-Be do processo de Gestão de Incidentes (GI) e do processo de Gestão de Problemas (GP) é superior ao da fase As-Is. Isto deve-se ao aumento do número de *Gateways* e consequentemente ao aumento do número de Actividades e de Conectores. Este aumento é comprovado através da métrica que mede o número total de objectos, onde podemos notar que o processo GI tem 27 elementos na Fase As-Is e 44 na fase To-Be, e o processo GP tem 14 elementos na fase As-Is e 31 elementos na fase To-Be.

		Nº de Caminho Possíveis (StartI, EndI)	Nº de End Events	Nº de TextAnnotations	Métrica HPC - difficulty	Métrica CNC	Métrica HKM
Gestão Incidentes	As-Is	3	1	0	1,5	1,1	12
	To-Be	10	1	0	1,5	1,2	15
Gestão Problemas	As-Is	1	1	0	1	0,923077	10
	To-Be	4	1	0	1,5	1,1	13

Tabela 6.40: Métricas da Gestão de Incidentes e Gestão de Problemas

Como podemos observar pela tabela 6.40 todas as métricas To-Be têm valores iguais ou superiores aos valores das métricas As-Is, ou seja, os processos To-Be têm definido um fluxo de actividade mais completo e pormenorizado, e por isso, apresentam um grau de complexidade superior. De seguida estudamos cada métrica separadamente:

- **Nº de Caminhos possíveis (StartI, EndI):** Esta métrica calcula o número de caminhos independentes. Através da tabela 6.40 podemos constatar que o processo de Gestão de Incidentes na fase As-Is tem 3 caminhos possíveis e na fase To-Be tem 10 caminhos possíveis. No processo Gestão de Problemas também sucede o mesmo (As-Is: 1 caminho, To-Be: 4 caminhos). Portanto, o número de caminhos que têm de ser considerados aquando da criação e desenvolvimento de um processo na fase To-Be é superior à fase As-Is. Podemos concluir então que a fase To-Be, de ambos os processos, é mais complexa uma vez que o criador do processo tem de lidar com todos os estados entre os fluxos de controlos construídos (*splits*) e as suas actividades e transições associadas.
- **Nº de EndEvents:** Esta métrica calcula o número de eventos finais. Através da tabela 6.40, podemos constatar que em ambos os processos, tanto na fase As-Is, como na fase To-Be, só existe um *endEvent*. Isto quer dizer que não existem sub-processos, ou seja, é apenas um único processo.
- **Nº de TextAnnotations:** Os objectos “*TextAnnotation*” não são usados da definição dos processos de Gestão de Incidentes e Gestão de Problemas.
- **Métrica HPC – difficulty:** Esta métrica determina uma medida quantitativa da complexidade baseada na compreensão de um BPMN, estimando a dificuldade dos processos (ver capítulo 5, secção 5.5.1.2 Complexidade do Processo/Modelo). Como podemos observar na tabela 6.40, o processo de Gestão de Incidentes apresenta valores iguais para a dificuldade do processo. Isto acontece, porque

esta métrica é calculada através da existência de actividades, *splits* e *joins* que estão presentes tanto na fase As-Is como na fase To-Be. Já no processo de Gestão de Problemas, a fase As-Is apresenta valores inferiores, pois apenas contém actividades e *splits*. A fase To-Be tem valores superiores, pois comporta actividades, *splits* e *joins*.

- **Métrica CNC - *Coefficient of Network Complexity***: Esta métrica é facilmente calculada com o número de arcos divididos pelo número de nós (actividades, eventos, *joins* e *splits*) (ver capítulo 5, secção 5.5.1.2 Complexidade do Processo/Modelo). Pela tabela 6.40 observamos que em ambos os processos, esta métrica apresenta valores superiores na fase To-Be. Isto é justificado através da quantidade de *FlowObjects* (GI: To-Be=23, As-Is=17. GP: To-Be=19, As-Is=13) e de Conectores (GI: To-Be=27, As-Is=18. GP: To-Be=20, As-Is=13) que aumenta da fase As-Is para a fase To-Be.

Através destas duas métricas HPC-*difficulty* e CNC, podemos concluir que a fase To-Be tanto do Processo de Gestão de Incidentes, como do processo de Gestão de Problemas é mais complexa que a fase As-Is.

- **Métrica *Henry and Kafura Metric (HKM)***: Ao dividirmos um BPM em sub-modelos modulares, podemos aumentar a sua compreensibilidade e, também, levar a modelos menores e reutilizáveis para futura manutenção (ver capítulo 5, secção 5.5.1.4 Modularização do Processo/Modelo). Esta métrica é definida com base no número total de actividades e no número de *StartEvents* e de *EndEvents*. Ambos os processos, nas suas duas fases só têm um *StartEvent* e um *EndEvent*. Adicionando ao facto de não existirem subprocessos, o aumento do número de actividades da fase As-Is para a fase To-be, podemos concluir, olhando para a tabela 6.40 (GI: To-Be=15, As-Is=12. GP: To-Be=13, As-Is=10), que a fase To-Be do processo de Gestão de Incidentes e do processo de Gestão de Problemas é mais complexa.

Na secção 6.4 apresentamos um método/técnica capaz de repetir o procedimento descrito nesta secção, de modo, a conseguirmos determinar o impacto da governança nos processos.

6.3.4 Conclusões do Caso de Estudo

Através da análise feita a este organismo público podemos caracterizá-lo da seguinte forma:

Governança de TI	
Perfil de governança de TI	Monarquia de negócio (o director é o líder)
Mecanismos da governança de TI	<ul style="list-style-type: none"> - Conselho de administração, - Comissão de aprovação do orçamento, - Equipas de processos com membros de TI, - Portais <i>web</i> e/ou intranets para suporte de governança de TI, - Mecanismo de rastreio/acompanhamento dos projectos de TI e dos recursos gastos em TI.
Complexidade do processo de Gestão de Incidentes e Gestão de Problemas	
Tamanho do processo	Após a “implementação” segundo as boas práticas, o grau de maturidade aumentou
Complexidade do processo	Após a “implementação” segundo as boas práticas, o grau de maturidade aumentou
Estrutura do processo	Após a “implementação” segundo as boas práticas, o grau de maturidade aumentou
Modularização do processo	Após a “implementação” segundo as boas práticas, o grau de maturidade aumentou

Tabela 6.41: Quadro síntese das características do caso de estudo

Apenas com um caso de estudo não nos é possível afirmar que existe uma relação entre a governança de TI de uma empresa e a complexidade dos seus processos de TI. Ou seja, não é possível afirmar que uma empresa com um perfil de monarquia do negócio, e com um desempenho bastante elevado encontra os seus processos de TI normalizados, documentados e amplamente postos em prática em toda a organização.

Unicamente, neste caso de estudo, e porque conhecemos o organismo, podemos concluir que o alto desempenho deste organismo se deve ao facto da gestão de topo interagir bastante com as unidades operacionais e preocupar-se em fazer comunicações transversalmente para toda a organização. Assim, a gestão de topo, consegue que os processos estejam bem definidos e integrados e sejam bem-postos em prática em todo o Organismo. A responsabilidade por cada processo encontra-se definida e é apoiada por formação formal. Todas as partes envolvidas nos processos estão conscientes dos riscos e das oportunidades potencialmente criadas. O aperfeiçoamento dos processos tem por base, sobretudo questões quantitativas e técnicas de suporte à TI, sendo possível seguir e medir a observância dos processos e os parâmetros que lhes estão associados. Os responsáveis pela gestão também definiram tolerâncias operacionais dos processos. Quando os processos parecem não funcionar eficazmente, ou com eficiência, são muitas vezes tomadas medidas. Os processos são por vezes aperfeiçoados e põem-se em uso as melhores práticas internas. A análise de causas profundas está a ser normalizada. A problemática do aperfeiçoamento continuo está a começar a ser abordada.

6.4 Impacto da Governança nos Processos

Para concluir a nossa investigação do objectivo a que nos tínhamos proposto no início desta dissertação, seria necessário que tivéssemos constituído uma amostra de casos de estudo (idealmente mais do que 30) onde tivesse sido possível repetir o procedimento que descrevemos na secção anterior.

Tal não aconteceu porque a empresa que se comprometeu a disponibilizar esta amostra não conseguiu, reunir e fornecer esta informação.

Não obstante, iremos descrever como iríamos proceder, caso dispuséssemos da referida amostra.

1. Determinar o perfil de governança de TI das empresas, caracterizado por um conjunto de variáveis que traduzem os domínios de decisão (Princípios de TI, Arquitectura de TI, Infraestrutura de TI, Necessidades de aplicações de negócio e Investimentos de TI), todas elas definidas numa mesma escala nominal de 6 categorias (Monarquia do Negócio, Monarquia de TI, Feudal, Federal, Duopólio e Anarquia).
2. Ver a influência que o perfil de governança de TI, determinado no ponto 1, tem em cada uma das 6 métricas identificadas na análise de correlação no início deste capítulo.

- Para isso, teríamos que fazer um teste de hipóteses, onde o nosso desenho experimental seria:

Variáveis independentes (vi) – Princípios de TI, Arquitectura de TI, Infraestrutura de TI, Necessidades de aplicações de negócio e Investimentos de TI.

Variável dependente (vd) – Cada uma das 6 métricas identificadas na análise de correlação.

- Depois de identificadas as vi's e a vd, teríamos que definir qual a hipótese que estamos a testar:

H_{0A} – O perfil de governança de TI não tem influência na complexidade do processo descrito pela métrica A

H_{1A} – O perfil de governança de TI tem influência na complexidade do processo descrito pela métrica A

Teremos tantos testes, como o número de métricas, ou seja, no total iremos ter 6 testes (métricas de A a F).

- O próximo passo, seria definir o tipo de teste que iríamos usar. Para tal é necessário:
 1. Determinar o tipo de escala da variável dependente
 No nosso estudo iremos ter dois tipos de escala: “*Absolute*” e “*Ratio*”
 “*Absolute*” para as métricas com valores numéricos inteiros, por exemplo, nº de *EndEvents*. “*Ratio*” para as métricas com valores numéricos fraccionários, por exemplo, métrica CNC.
 2. Verificar se a variável dependente segue uma distribuição normal (paramétrica ou não-paramétrica). Esta verificação pode ser feita através de

testes de hipóteses que avaliam a normalidade, como por exemplo, o teste de *Kolmogorov-Smirnov*⁴⁷.

2.1 Se a variável dependente seguir uma distribuição normal então, tendo em conta o nosso desenho experimental (5 vi's numa escala nominal com 6 categorias), podemos utilizar o teste paramétrico da ANOVA⁴⁸ factorial.

2.2 2.2 Se a variável dependente não seguir uma distribuição normal então, tendo em conta o nosso desenho experimental (5 vi's numa escala nominal com 6 categorias), podemos utilizar o teste da ANOVA⁴⁹ factorial não-paramétrica.

⁴⁷ Para informações adicionais sobre este teste consultar, por favor, http://pt.wikipedia.org/wiki/Teste_Kolmogorov-Smirnov

⁴⁸ Para mais informações sobre o teste da ANOVA consultar http://en.wikipedia.org/wiki/Analysis_of_variance

⁴⁹ Para mais informações sobre o teste da ANOVA consultar http://en.wikipedia.org/wiki/Analysis_of_variance

[Esta página foi intencionalmente deixada em branco]

Capítulo 7

Conclusões e Trabalho Futuro

Conteúdo

7 CONCLUSÕES E TRABALHO FUTURO.....	120
7.1 CONCLUSÕES.....	120
7.2 EVOLUÇÃO FUTURA	125

Neste capítulo são apresentadas as conclusões desta dissertação e de que forma foram atingidas as contribuições previstas. Esta dissertação termina com a identificação das linhas de orientação gerais para a continuação futura do trabalho desenvolvido.

7 Conclusões e Trabalho Futuro

7.1 Conclusões

Nesta secção vamos enumerar, apresentar e recapitular as contribuições previstas nesta dissertação, de que forma foram atingidas e quais as dificuldades sentidas para as alcançar:

1. Determinar o perfil de governança de TI de uma organização

Método: Para determinarmos o perfil de governança de TI elaboramos um questionário que contém 12 perguntas de resposta rápida e está dividido em 3 secções: A - Elementos de Governança de TI, B - Desempenho da Governança de TI e C - Contexto Organizacional. Este questionário foi baseado em trabalho de investigação desenvolvido na *Sloan School of Management* do MIT, nomeadamente o descrito no artigo de Hoffman e Weill - “*Banknorth: Designing IT Governance for a Growth-Oriented Business Environment*” [Hoffman and Weill, 2007].

Resultados: Com os resultados obtidos através dos questionários, pudemos analisar e mapear os tipos de decisões existentes numa empresa, os arquétipos para a tomada dessas decisões e como é que essas decisões são tomadas e monitorizadas, através de um conjunto de mecanismos de governança.

Concluimos que as empresas constantes na amostra estão categorizadas em dois tipos de perfis.

Para os domínios de decisão: (i) Princípios de TI; (iii) Infraestrutura de TI; e (v) Investimentos de TI, ambos os perfis seguem o arquétipo “**Monarquia do Negócio**” ou seja, os directores (CEO, COO, CFO, CHRO, CIO) é que lideram. Para o domínio de decisão: (iv) Necessidades de aplicações de negócio seguem o arquétipo “**Duopólio**”, ou seja, é governado pelos directores e pelas chefias (dependendo da estrutura organizacional, temos líderes de unidades de negócio ou outros líderes dos vários departamentos/secções que constituem a organização).

A diferença entre os dois perfis situa-se no domínio de decisão: (ii) Arquitectura de TI, onde um perfil segue o arquétipo “**Monarquia do Negócio**” e o outro perfil segue o arquétipo “**Monarquia de TI**”, ou seja, é liderado pelas chefias e pelos coordenadores (donos de processo de negócio que possuem responsabilidade e autoridade pela operação e melhoria do processo como um todo - e utilizadores finais chave, geralmente com algum tipo de chefia intermédia).

Com esta caracterização dos perfis de governança de TI e fazendo uma comparação com os perfis com os melhores tipos de desempenho (figura 3.6), identificados pelos citados investigadores americanos, posicionamos os perfis de governança das empresas portuguesas num intervalo de 60 a 80% de consonância com os três melhores perfis de governança.

No nosso estudo, também concluimos que as empresas portuguesas tendem a ser centralizadas na sua abordagem à governança de TI. As estratégias destas empresas enfatizam operações centradas em medidas do custo dos processos de negócio e da rentabilidade.

Pudemos concluir também, a partir deste estudo, que a média das empresas inquiridas apresenta um desempenho da governança de TI de 30%, ou seja, as estratégias adoptadas são pouco focadas, os objectivos para os investimentos de TI não são muito claros e eventualmente ocorrem muitas mudanças de ano para ano, mostrando a pouca estabilidade na governança.

Em termos de conclusão geral, de todos os dados que foram analisados nos inquéritos e de todos os artigos que foram estudados nesta área de domínio, nomeadamente através dos artigos de *Peter Weill*, podemos dizer que a ineficiência da governança de TI é um problema na nossa era e que se deve essencialmente:

- a) **à gestão de topo não ver valor nos investimentos de TI:** é originado pelas dúvidas e desconhecimento que os administradores seniores possuem quanto ao valor agregado da TI nos negócios das empresas (ou seja, os negócios das empresas só sobrevivem devido aos investimentos de TI que já foram efectuados). O problema surge porque, frente a esta dificuldade, o CEO acaba por abdicar desta responsabilidade, deixando que a orientação e monitorização dos investimentos de TI fiquem sobre a responsabilidade dos gestores de TI. Isto poderá conduzir a tomadas de decisões de TI erradas ou mesmo tomadas de decisões divergentes ao direccionamento de negócio da empresa;
- b) **à TI tornar-se uma barreira para novas implementações estratégicas da empresa:** a TI não caminha por si só, nem possui independência das decisões estratégicas do conselho executivo da empresa. Quando se trata de implementações de novas tecnologias ou mesmo de crescimento tecnológico, a área de TI carece do seu envolvimento com as restantes unidades de negócio da empresa. Isto é necessário pois, a área de TI mantém um “legado” que suporta os negócios da empresa. Se, por exemplo, numa estratégia executiva é decidido alterar a plataforma de TI, isto poderá gerar um impacto e um risco tão grande para o negócio que poderá impossibilitar a efectivação de um objectivo da empresa;
- c) **aos mecanismos para se tomar decisões serem lentos e contraditórios:** a governança de TI deve prover consistência para a Gestão de Projectos, pois a Gestão de Projectos assegura a alocação de recursos dedicados, com uma sequência disciplinada de realização, com etapas e com o histórico do avanço dos projectos. A Gestão de Projectos permite a visualização e avaliação, por parte dos executivos da empresa, da progressão dos mesmos. Se estes projectos não estiverem a seguir as orientações estratégicas da empresa como um todo, de nada adiantará serem implementados projectos com objectivos que não estejam alinhados com as estratégias de negócio;

Dificuldades: A maior dificuldade sentida nesta fase foi a recolha de dados, ou seja, conseguir obter, em tempo útil, uma amostra considerável de questionários preenchidos. Felizmente, este obstáculo foi superado com a cooperação do itSMF Portugal, em particular durante a realização da sua conferência anual que decorreu na Reitoria da UNL em Maio de 2009.

2. Calcular a complexidade de um processo

Método: Para determinarmos a complexidade de um processo de TI, usando como base a técnica de *Metamodel Driven Measurement* (M2DM), proposta em [Brito e Abreu, 2001], um conjunto de métricas usando a linguagem de restrições OCL sobre um metamodelo BPMN também desenvolvido na equipa de investigação QUASAR, que é descrito em [Freitas, 2010] e reproduzido no anexo B; Robustecemos este metamodelo com restrições (regras de boa formação) em OCL e instanciámo-lo com base num processo de elevada complexidade (vários milhares de meta-objects e meta-ligações), igualmente desenvolvido no grupo QUASAR no âmbito de outra dissertação [Costa, 2010].

As métricas definidas foram especificadas com base nos artigos “*Complexity Metrics for Measuring the Understandability and Maintainability of Business Process Models using Goal-Question-Metric (GQM)*” de Ghani e seus colegas [Ghani, 2008] e “*A Discourse on Complexity of Process Models*” de Jorge Cardoso e seus colegas [Cardoso, 2006]. Com base, essencialmente, nestes artigos, adoptámos 4 categorias de métricas: tamanho, complexidade, estrutura e modularização. Foram ainda propostas várias outras métricas definidas de raiz, baseadas no atravessamento do grafo orientado que representa o processo, com recurso a algoritmos recursivos. Por exemplo, conseguimos (i) calcular métricas dando uma origem e um destino ou dando apenas uma origem; (ii) calcular o maior e o menor caminho; (iii) calcular métricas para avaliar o modelo ou o processo; (iv) navegar nos processos com *drill-down* (afundando pelos subprocessos); (v) retornar apenas os *flowObjects* ou todos os elementos (incluindo *connectors* e *artifacts*), etc.

Resultados: Nesta fase, e com base nos artigos nesta área de domínio, examinámos várias contribuições da Engenharia de Software, da ciência cognitiva e da teoria dos grafos, e discutimos e implementámos um conjunto de métricas de complexidade de modelos de processo. Através de um estudo correlacional pudemos concluir que algumas das métricas definidas estão fracamente relacionadas, o que permitiu identificar um conjunto de métricas independentes que, como ilustrado e explicado no capítulo 6, podem ser utilizadas por analistas de processos de negócio e *designers* do processo para analisar várias facetas da complexidade de modelos do processo.

Dificuldades: Esta fase da dissertação foi bastante mais exaustiva do que inicialmente tinha sido previsto. No total, programámos 337 funções em OCL. Tal como na fase anterior, uma das maiores dificuldades sentidas nesta fase foi a recolha de dados reais. Em Portugal, são muito poucas as empresas que têm feito estudos com *workflows* das fases “As-Is” e “To-Be” de processos de gestão de serviços de TI. Para além disso, a qualidade dos dados da amostra é muito importante pois, pode constituir uma séria ameaça à validade dos resultados. Nos dados amostrais é necessário ter em conta (i) a transformação de dados, pois podem implicar um esforço muito grande para serem avaliados pelo USE, caso se encontrem em formatos difíceis de transformar, e (ii) o facto de não existir uma metodologia de desenho de processos com BPMN que seja aceite de forma universal, ou seja, podem existir modelos mais pobres que outros, dependendo da utilização ou não de, por exemplo, artefactos,

subprocessos, etc. Foi por estes motivos que tivemos que validar as métricas com um exemplo já transformado (num formato que o USE interprete) produzida no âmbito de outra dissertação.

3. Influência do modelo de governança na adopção baseada em processos

Método: Para estudarmos a influência do modelo de governança na adopção baseada em processos de TI nas organizações portuguesas, pusemos em prática o definido nos pontos 1 e 2, atrás referidos. Seguidamente, com base no artigo “*The IT organization modeling and assessment tool: Correlating IT governance maturity with the effect of IT*” [Simonsson and Johnson, 2008] analisámos um caso de estudo de um organismo público português.

Resultados: Através do ponto 1 e do caso de estudo analisado, constatámos que fazer o levantamento/caracterização dos componentes da governança de TI é simples, directo e comparável. É simples e directo porque, o método definido e descrito para caracterizar a governança de TI de uma empresa é baseado em apenas algumas questões que podem ser usadas para atribuir responsabilidades de TI de alto nível na tomada de decisão. Através das respostas a estas questões conseguimos traçar o perfil de governança de TI e compreender como a gestão de topo governa a empresa. É comparável, porque desde que os inquiridos respondam com sinceridade é possível fazer um estudo comparativo das respostas dadas.

Através do ponto 2 e do caso de estudo analisado, constatámos que calcular o grau de complexidade dos processos de TI pode não ser tão simples, directo e comparável. Primeiro, porque é necessário que as empresas compreendam a necessidade das boas práticas e implementem ITIL (*IT Infrastructure Library*) ou outro referencial que possa auxiliar a criação de processos relacionados com a entrega e suporte de serviços de TI. Segundo porque, como já referido, não existe uma metodologia exacta de desenho de processos com BPMN.

Por estes contratempos e os já referidos anteriormente, apenas conseguimos resposta de um organismo que reúne todas as condições necessárias para ser estudado. Este organismo está a trabalhar para a certificação ISO 20000 e já tem dois processos definidos segundo as boas práticas. Através deste caso de estudo não podemos concluir que existe uma relação entre o perfil de governança de TI e o grau de complexidade destes dois processos. Por isso, como resultado desta fase da dissertação, apenas podemos concluir que definimos, descrevemos, explicamos e demonstramos um método/técnica capaz de extrair o perfil de governança, o grau de complexidade dos seus processos de TI e a relação entre eles.

Dificuldades: Assim como nas fases anteriores, a maior dificuldade sentida nesta fase foi a recolha de dados reais. Foi por este motivo que não conseguimos validar este método/técnica cujo objectivo último era testar a hipótese de relação causa/efeito entre governança e gestão de processos aplicada às TI.

Testemunho

Este processo de aprendizagem exigiu muito trabalho, paciência e determinação.

Foi um percurso individual algo penoso pois, tive que dividir-me entre a investigação, a minha actividade profissional e a minha vida pessoal mas, também foi um percurso muito enriquecedor, gratificante e uma oportunidade excepcional de crescimento académico e também pessoal. Ao longo deste percurso, tive a sorte de caminhar ao lado de um excelente e notável orientador, com um espírito de grande incentivo à investigação, sempre disponível e sempre com uma palavra amiga que transmitiam entusiasmo e coragem; e ao lado de alguns colegas excepcionais do QUASAR, com uma excelente dinâmica de grupo, o que contribuiu imenso para o estudo e discussão dos temas no desenvolvimento dos trabalhos. O companheirismo esteve sempre presente e nunca faltou encorajamento nas horas mais difíceis.

Do ponto de vista metodológico aprendi imenso. Aprendi a organizar as ideias, a saber o que interessa e o que não interessa, a classificar o conhecimento, a perder o medo de ir às fontes e às sínteses, a estudar e interpretar o estado da arte, a formatação científica, etc. Aprendi que a bibliografia necessária a um bom desempenho obriga a um trabalho contínuo e diário que foi de algum modo inesperado, mas garantiu a qualidade da aprendizagem.

O tempo é sempre um factor crítico neste tipo de trabalho. Contudo, graças a orientação que tive, a calendarização permitiu-me não fazer nada à pressa e ter liberdade absoluta para caminhar ao meu ritmo.

Esta dissertação também me permitiu fazer uma ponte entre a minha experiência profissional, baseada em projectos ITIL, ISO 20000 e ISO 27001 com a investigação académica.

Depois de concluir a licenciatura, optei por trabalhar na área de gestão de serviços de TI. Tive formação profissional, onde aprofundi conhecimentos em ITIL, ISO 20000 e ISO 27001. No entanto, não achei suficiente e senti que precisava de uma abordagem diferente, entendi que necessitava de melhorar os meus conhecimentos nesta área. Decidi fazer este mestrado com o apoio da empresa onde trabalho actualmente. Foi uma boa estratégia, que me permitiu dedicar mais tempo a esta investigação.

Assim, do ponto de vista profissional esta dissertação ajudou-me muito. Entendi conceitos que estavam de alguma forma pouco claros para mim, entendi a importância da governança de TI, aprendi técnicas de construção de inquéritos que serão muito úteis em futuros projectos onde será necessário fazer levantamento/caracterização de informação, alarguei os meus horizontes, vi a importância que este tema tem fora das nossas fronteiras, constatando que as boas práticas são um tema quente na actualidade. Em suma, consolidei o meu conhecimento nesta área.

Toda esta experiência e conhecimento académico fez-me sentir mais a vontade com a área de governança de TI e conseqüente mais confiante, que é uma característica essencial para o bom desempenho da minha função como consultora em ITIL, ISO 20000 e ISO 27001.

Dos aspectos que julgo dever destacar, como essenciais para o sucesso de um aluno de mestrado, saliento os seguintes:

- Grande motivação para iniciar um processo estimulante, onde se encontrará sozinho e no qual terá de gerir o seu tempo, os seus obstáculos e as suas frustrações;
- Capacidade de gerar empatias e amizades, como forma de aumentar a colaboração entre colegas e diminuir a potencial desmotivação provocada por situações inesperadas;
- Genuína vontade de aprender, em regime de liberdade de ritmos, tempos e escala de profundidade dos temas.

Por tudo isto, este mestrado foi, no mínimo, intenso e estimulante. Uma grande experiência que recomendo, onde no fim o nosso trabalho é reconhecido.

7.2 Evolução Futura

O propósito do caso de estudo analisado não foi a validação do método/técnica proposto nesta dissertação pois, um único exemplo não é suficiente para uma validação. Assim, o seu propósito foi a exemplificação da aplicação deste método/técnica. Desta forma é proposto, como continuação ao trabalho desenvolvido, a aplicação deste método/técnica a uma amostra significativa de empresas, por forma a validar a possível relação existente entre o perfil de governança de TI e o grau de maturidade dos processos de TI.

A avaliação da maturidade implica uma análise semântica e levantamento sistemático relativo à adopção de uma série de práticas / actividades recomendadas. Não conseguimos determinar uma maturidade de um processo apenas com as métricas que possuímos. Mas acreditamos que elas expliquem parcialmente essa maturidade. Processos mais maduros implicam mais actividades, mais decisões e por isso certamente uma maior complexidade. Há testes estatísticos como o teste da ANOVA que permitem determinar, por exemplo, qual é a variância explicada, neste caso da maturidade, que se consegue a partir das métricas de complexidade. Mas para o aplicarmos precisávamos de ter diagramas de processos cuja maturidade tivesse sido previamente determinada com abordagens próprias (ex: CMMI).

Assim, para tornar este estudo mais completo poder-se-ia associar a complexidade de um processo calculada através das métricas a um modelo de maturidade de capacidades como o CMMI. Desta forma poderíamos saber exactamente em que grau de maturidade estamos, para que grau de maturidade evoluímos e se estamos a melhorar. Ou seja, utilizaríamos uma pontuação do nível de maturidade atingido na realização de cada objectivo estipulado pela gestão.

Outro melhoramento seria reunir ou estipular *Key Performance Indicators* (KPIs) que estivessem relacionados, de alguma forma, com os processos de gestão de TI e com as métricas definidas pois, para que as empresas possam suportar a governança de TI, há necessidade de se ter uma visão do que está a ser implementado e da sua maturidade. Deve ser feito um acompanhamento sobre o desempenho

dos processos que suportam os negócios da empresa e assegurar que as orientações corporativas serão alcançadas, ou seja, deve ser verificado que os processos de negócio estão realmente a seguir as orientações dadas pela gestão de topo. Estes acompanhamentos e verificações devem ser baseados em relatórios gerados pelos sistemas de informações pois, sem estes relatórios, os gestores, não têm condições de avaliar a empresa de uma forma mais precisa, nem conseguem tomar decisões assertivas. Isto iria trazer inúmeras vantagens para as empresas, nomeadamente, poderiam maximizar benefícios e capitalizar oportunidades e vantagens competitivas.

Por exemplo, se uma empresa utiliza o ITIL como guia de referência, então os seus processos podem ser implementados por etapas. As métricas iriam avaliar a complexidade de um processo e os KPIs iriam avaliar se os processos estão a ser executados de acordo com as expectativas. Assim, já teríamos condições de avaliar a qualidade dos processos, que por sua vez é a base para a contínua optimização e aperfeiçoamento da concepção do processo.

Como prometido, também é nosso objectivo apresentar publicamente os resultados globais do questionário da governança de TI num workshop do itSMF Portugal, pois assim, desta forma, também estaremos a sensibilizar os gestores de empresas para a importância da governança de TI.

Anexo A

Inquérito Governança de TI

Conteúdo

A. INQUÉRITO DE GOVERNANÇA DE TI.....	128
A) INQUÉRITO SOBRE GOVERNANÇA DE TI	128
B) RESULTADOS DO INQUÉRITO SOBRE GOVERNANÇA DE TI	138
C) INTERPRETAÇÃO DOS RESULTADOS DO INQUÉRITO SOBRE GOVERNANÇA DE TI.....	158

Neste anexo é apresentado o inquérito de governança de TI realizado, os seus resultados e interpretação dos mesmos.

A. Inquérito de Governança de TI

a) Inquérito sobre governança de TI

ENQUADRAMENTO:

Este inquérito, realizado com o apoio do itSMF, tem como intuito ajudar a compreender os arquétipos de governança de TI nas organizações em Portugal. Em particular, pretende-se realizar um levantamento / caracterização dos componentes da governança de TI, como é que eles podem ser representados e o que se faz para se obter uma governança de TI eficaz.

A sua resposta a este inquérito, que se dirige a profissionais de organizações (ou sectores destas), cuja principal actividade se centra nas TIs, é extremamente valiosa. Retribuiremos a sua participação, enviando-lhe os resultados deste estudo por correio electrónico. Há um campo opcional neste inquérito onde poderá fornecer um endereço de e-mail, caso o pretenda.

A investigação subjacente a este estudo está a decorrer no âmbito de uma dissertação de mestrado em Engenharia Informática, na Faculdade de Ciências e Tecnologia, da Universidade Nova de Lisboa. A construção deste inquérito baseou-se em trabalho de investigação desenvolvido na *Sloan School of Management* do MIT.



ESTRUTURA:

Este inquérito, com duração aproximada de preenchimento de 15 minutos, contém 12 perguntas de resposta rápida e está dividido em 3 secções: Elementos de Governança de TI, Desempenho da Governança de TI e Contexto Organizacional.

Dado que não se trata de um teste, não há respostas certas ou erradas. Agradecemos que preenchesse o questionário tão completamente quanto possível, apelando à veracidade e sinceridade na sua resposta.

O anonimato e a confidencialidade dos dados individuais fornecidos estão absolutamente garantidos. Só iremos apresentar publicamente os resultados globais, o que será efectuado num Workshop do itSMF a anunciar.

A sua colaboração é indispensável para o sucesso deste estudo, pelo que agradecemos desde já a sua resposta.

A. Elementos de Governança de TI**1. Domínio de Decisão de TI**

As organizações tomam decisões sobre 5 principais áreas de TI, chamadas de “domínios de decisão” e que são os seguintes:

- i. Investimento de TI,
- ii. Princípios de TI,
- iii. Estratégia de Infraestrutura de TI,
- iv. Arquitectura de TI e Standards e,
- v. Necessidade de aplicações de negócio.

Para cada um dos domínios de decisão de TI existem grupos que contribuem para as decisões ou que efectivamente decidem:

Directores: Director Geral (CEO – *Chief Executive Officer*), Director de Operações (COO - *Chief Operating Officer*), Director Financeiro (CFO - *Chief Financial Officer*), Director dos Recursos Humanos (CHRO – *Chief Human Resources Officer*), Director de Informática (CIO - *Chief Information Officer*), etc. Os directores têm geralmente assento no Conselho de Administração.

Chefias: Dependendo da estrutura organizacional, temos líderes de unidades de negócio ou outros líderes dos vários departamentos/secções que constituem a organização.

Coordenadores: Donos de processo de negócio - possuem responsabilidade e autoridade pela operação e melhoria do processo como um todo - e Utilizadores finais chave, geralmente com algum tipo de chefia intermédia

Observe cada um dos domínios de decisão de TI abaixo e depois, indique com uma cruz, os grupos que **contribuem para as decisões** (C) ou que **efectivamente decidem** (D). Atenção: Para cada grupo pode seleccionar uma ou ambas as opções (C e/ou D).

Qual o nível de intervenção (Contribui ou Decide) de cada um dos seguintes actores relativamente:

a) Ao investimento em TI ou em negócios que dependam grandemente das TIs

Directores		Chefias		Coordenadores		Não Aplicável
C	D	C	D	C	D	

- b) À definição do papel das TIs no negócio da organização (por exemplo: suportar cortes nos custos ou crescimento nas receitas)

Directores		Chefias		Coordenadores	
C	D	C	D	C	D

Não Aplicável

- c) Às estratégias de implementação/evolução da infraestrutura de TI (por exemplo: redes, hardware, etc.)

Directores		Chefias		Coordenadores	
C	D	C	D	C	D

Não Aplicável

- d) Às opções técnicas relativamente à arquitectura de TI (por exemplo: opção por arquitectura cliente/servidor, web service, etc.)

Directores		Chefias		Coordenadores	
C	D	C	D	C	D

Não Aplicável

- e) À necessidade de aquisição ou desenvolvimento de novas aplicações para suportar o negócio (portfolio aplicacional)

Directores		Chefias		Coordenadores	
C	D	C	D	C	D

Não Aplicável

Nas próximas 4 questões, considere que a proximidade de cada um dos extremos do contínuo é indicador do seu nível de identificação dessa eficiência, importância ou grau de sucesso.

Se o aspecto em estudo não for usado na sua organização, assinale com uma cruz na coluna “Não Aplicável”.

2. Órgãos com intervenção na governança de TI

Nas organizações existem, com frequência, uma variedade de órgãos com maior ou menor intervenção na governança de TI.

Indique seguidamente a avaliação que faz da sua eficiência desses órgãos para potenciar o valor do negócio das TIs na sua organização:

2.1 Conselho de Administração (constituído maioritariamente pelos directores dos vários sectores da organização)

	--	-	±	+	++				
INEFICIENTE						ALTAMENTE EFICIENTE	<table><tr><td>Não Aplicável</td><td></td></tr></table>	Não Aplicável	
Não Aplicável									

2.2 Comissão de aprovação do orçamento

	--	-	±	+	++				
INEFICIENTE						ALTAMENTE EFICIENTE	<table><tr><td>Não Aplicável</td><td></td></tr></table>	Não Aplicável	
Não Aplicável									

2.3 Conselho de gestão de TI (órgão misto composto por executivos de TI e de negócio)

	--	-	±	+	++				
INEFICIENTE						ALTAMENTE EFICIENTE	<table><tr><td>Não Aplicável</td><td></td></tr></table>	Não Aplicável	
Não Aplicável									

2.4 Conselho de administração de TI (órgão exclusivamente composto por executivos de TI)

	--	-	±	+	++				
INEFICIENTE						ALTAMENTE EFICIENTE	<table><tr><td>Não Aplicável</td><td></td></tr></table>	Não Aplicável	
Não Aplicável									

2.5 Conselho da arquitectura de TI

	--	-	±	+	++				
INEFICIENTE						ALTAMENTE EFICIENTE	<table><tr><td>Não Aplicável</td><td></td></tr></table>	Não Aplicável	
Não Aplicável									

2.6 Equipas de processos com membros de TI

	--	-	±	+	++				
INEFICIENTE						ALTAMENTE EFICIENTE	<table><tr><td>Não Aplicável</td><td></td></tr></table>	Não Aplicável	
Não Aplicável									

2.7 Relacionamento de gestores de negócio / gestores de TI

	--	-	±	+	++				
INEFICIENTE						ALTAMENTE EFICIENTE	<table><tr><td>Não Aplicável</td><td></td></tr></table>	Não Aplicável	
Não Aplicável									

3. Mecanismos de Governança de TI

Nas organizações existem, com frequência, uma variedade de mecanismos de governança de TI.

Indique seguidamente a avaliação que faz da eficiência desses mecanismos para potenciar o valor do negócio das TIs na sua organização:

3.1 Mecanismo de rastreio/avaliação do valor de negócio da TI

	--	-	±	+	++		
INEFICIENTE						ALTAMENTE EFICIENTE	Não Aplicável

3.2 Portais web e/ou intranets para suporte de governança de TI

	--	-	±	+	++		
INEFICIENTE						ALTAMENTE EFICIENTE	Não Aplicável

3.3 Mecanismo de rastreio/acompanhamento dos projectos de TI e dos recursos gastos em TI

	--	-	±	+	++		
INEFICIENTE						ALTAMENTE EFICIENTE	Não Aplicável

3.4 Existência de Acordos de Nível de Serviço (SLA) de TI

	--	-	±	+	++		
INEFICIENTE						ALTAMENTE EFICIENTE	Não Aplicável

3.5 Existência de Acordos internos de recuperação de custos de TI

	--	-	±	+	++		
INEFICIENTE						ALTAMENTE EFICIENTE	Não Aplicável

B. Desempenho da governança de TI**1. Objectivos de governança de TI**

Em seu entender, qual tem sido (no passado) a influência da governança de TI da sua organização relativamente aos seguintes aspectos.

1.1 Boa rentabilização do investimento em TI

	--	-	±	+	+		
POUCO IMPORTANTE						MUITO IMPORTANTE	Não Aplicável

1.2 Uso eficaz da TI para melhorar a utilização dos recursos (humanos e materiais) da organização

	--	-	±	+	+		
POUCO IMPORTANTE						MUITO IMPORTANTE	Não Aplicável

1.3 Uso eficaz da TI para o crescimento do negócio

	--	-	±	+	+		
POUCO IMPORTANTE						MUITO IMPORTANTE	Não Aplicável

1.4 Uso eficaz da TI para conceber maior flexibilidade ao negócio

	--	-	±	+	+		
POUCO IMPORTANTE						MUITO IMPORTANTE	Não Aplicável

2. Resultados de governança de TI

Em seu entender, qual é a importância que a governança de TI da sua organização deveria dar (no futuro) aos seguintes aspectos:

2.1 Boa rentabilização do investimento em TI

	--	-	±	+	++		
INSUCESSO						MUITO SUCESSO	Não Aplicável

2.2 Uso eficaz da TI para melhorar a utilização dos recursos (humanos e materiais) da organização

	--	-	±	+	++		
INSUCESSO						MUITO SUCESSO	Não Aplicável

2.3 Uso eficaz da TI para o crescimento do negócio

	--	-	±	+	++		
INSUCESSO						MUITO SUCESSO	Não Aplicável

2.4 Uso eficaz da TI para conceber maior flexibilidade ao negócio

	--	-	±	+	++		
INSUCESSO						MUITO SUCESSO	Não Aplicável

C. Contexto Organizacional

1. Por favor, indique com uma ou mais cruzes, qual é o sector de actividade da sua organização:

Agricultura/Pescas		Indústria	
Ambiente		Militar	
Consultoria		Organização sem fins lucrativos	
Cultura		Saúde	
Desporto		Sector Financeiro	
Educação		Software House	
Energia		Utilities (água, energia, Telecom)	
Governo		Outra: _____	

2. A que tipo de organização/sector pertence:

Prestadora de serviços de TI internamente	
Prestadora de serviços de TI externamente	
Consultora integradora	
Outra: _____	

3. Aproximadamente, que percentagem de custos anuais da organização, é dedicada a TI?

4. Dependendo do tipo de estrutura da sua organização, por favor, indique:

a. O número de unidades de negócio independentes?

b. O número de direcções independentes?

5. Qual o cargo que ocupa na organização?

6. Por favor, indique o total aproximado de profissionais de TI na sua organização:

7. Por favor, indique o total aproximado de colaboradores da sua organização:

8. Vamos elaborar um relatório com base na análise das respostas dadas a este inquérito. Se o pretender receber, indique neste campo um e- mail para onde lho possamos enviar

Mais uma vez gostaríamos de agradecer o preenchimento deste questionário.

Estamos certos de que compreendeu a importância deste estudo na identificação de arquétipos de governança de TI nas empresas portuguesas.

Se quiser saber mais sobre este estudo, ou estiver interessado em dar alguma sugestão ou informação adicional, por favor contacte Raquel Porciúncula, através do endereço electrónico porci@netcabo.pt.

A realização com sucesso deste estudo só é possível graças ao apoio empenhado do [itSMF Portugal](#), que aqui reconhecemos publicamente.

De novo agradecemos a vossa colaboração e apresentamos os melhores cumprimentos,

[Grupo de investigação QUASAR](#)

Faculdade de Ciências e Tecnologia

Universidade Nova de Lisboa

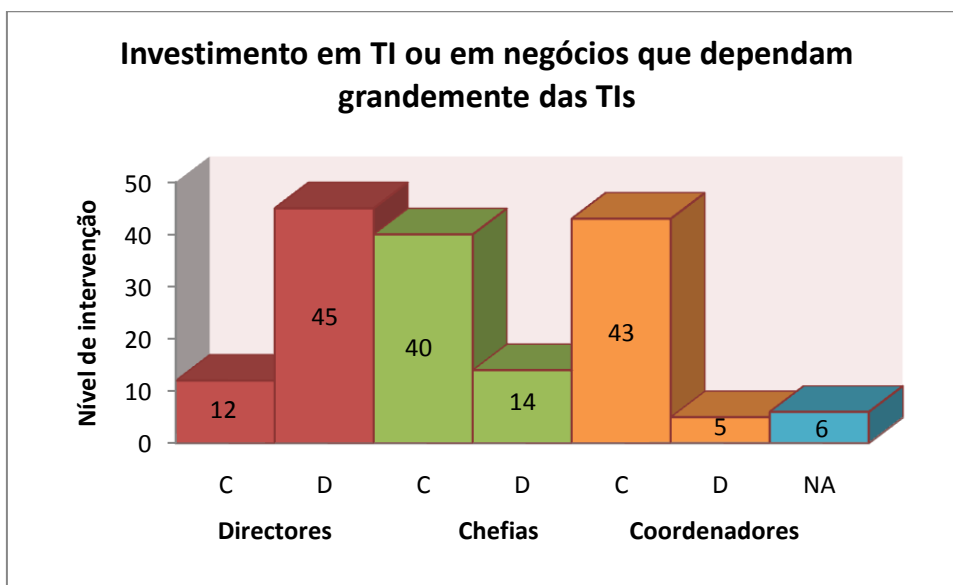
b) Resultados do inquérito sobre governança de TI

A. Elementos de Governança de TI

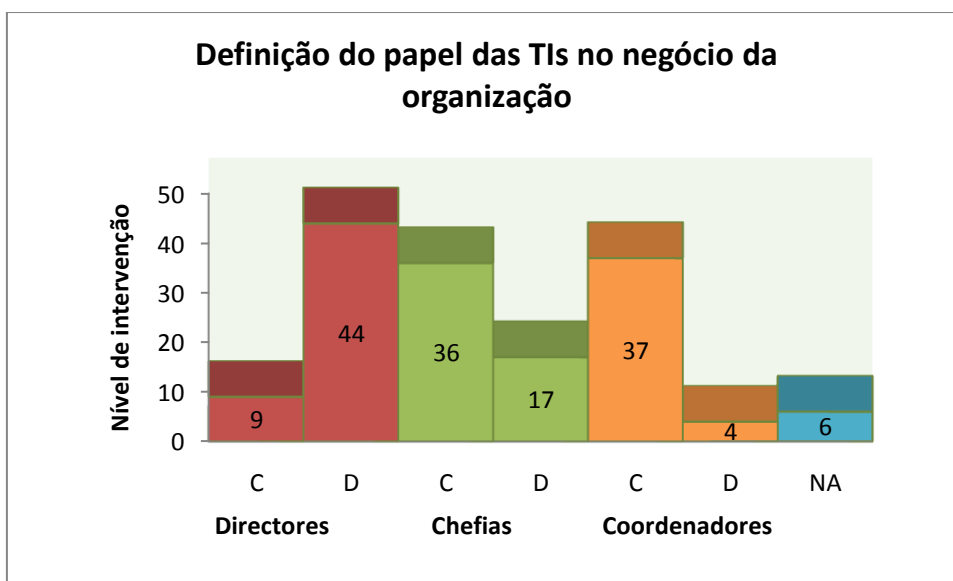
1. Domínio de Decisão de TI

Nível de intervenção (Contribui ou Decide) de cada um dos seguintes actores relativamente a:

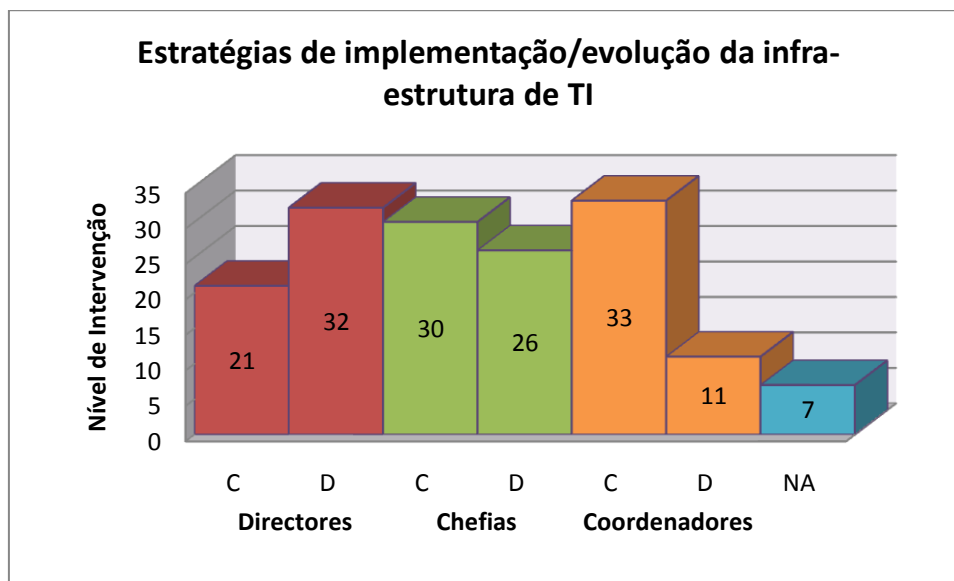
a)



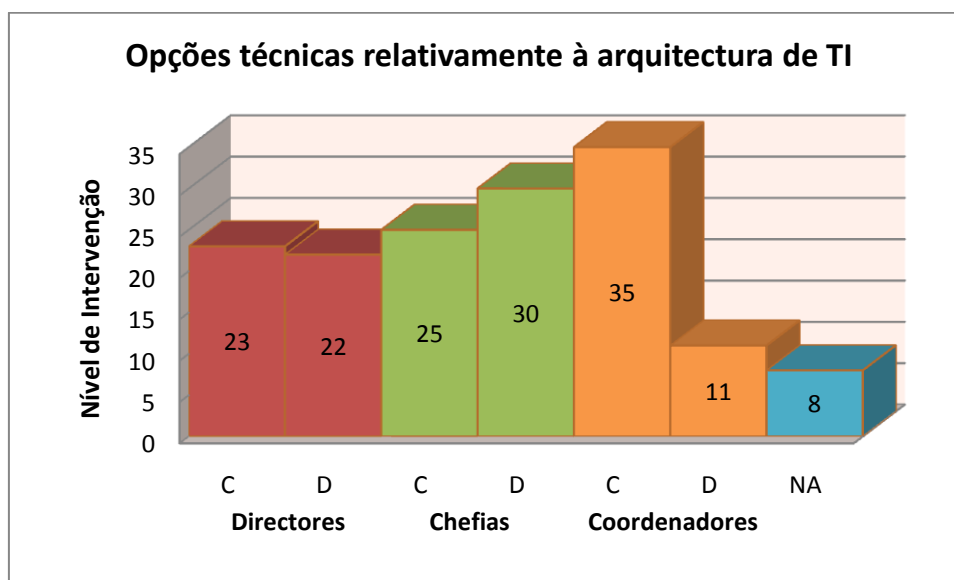
b)



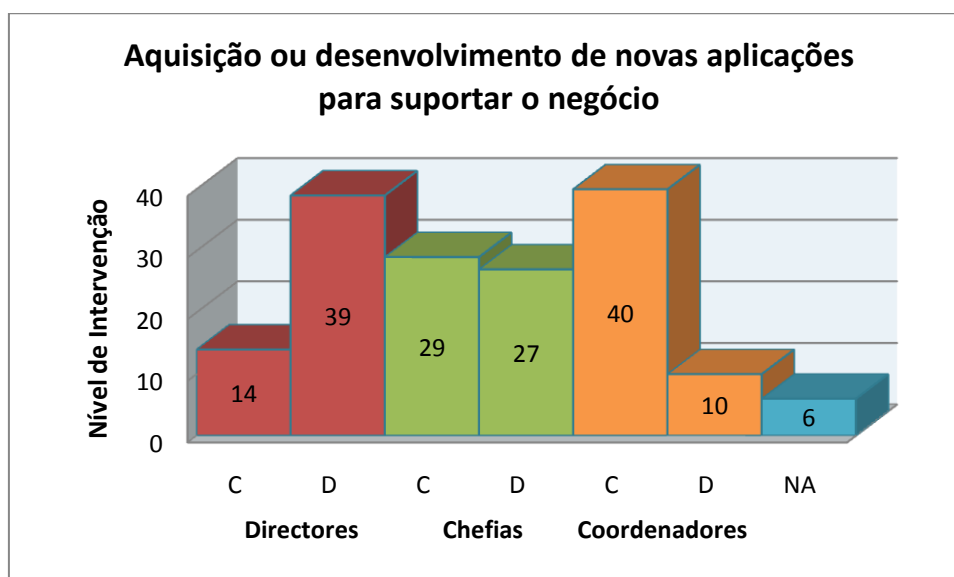
c)



d)



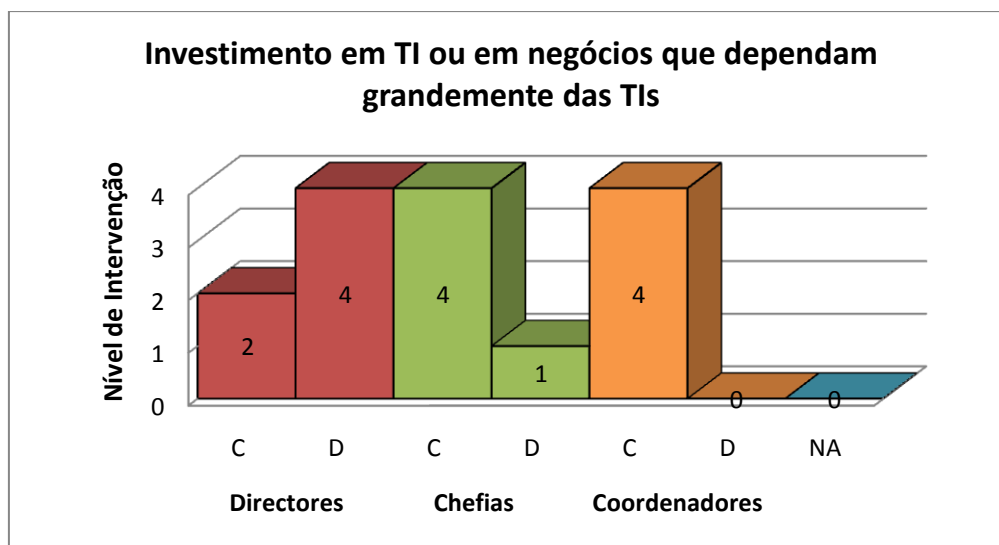
e)



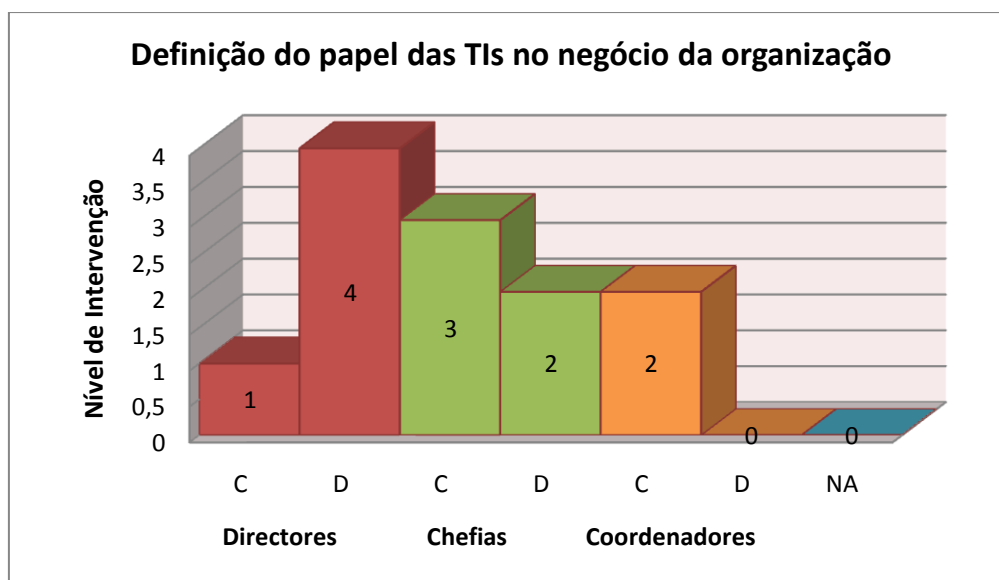
➤ **Pequenas Empresas** (< 50 colaboradores)

1. Nível de intervenção (Contribui ou Decide) de cada um dos seguintes actores relativamente a:

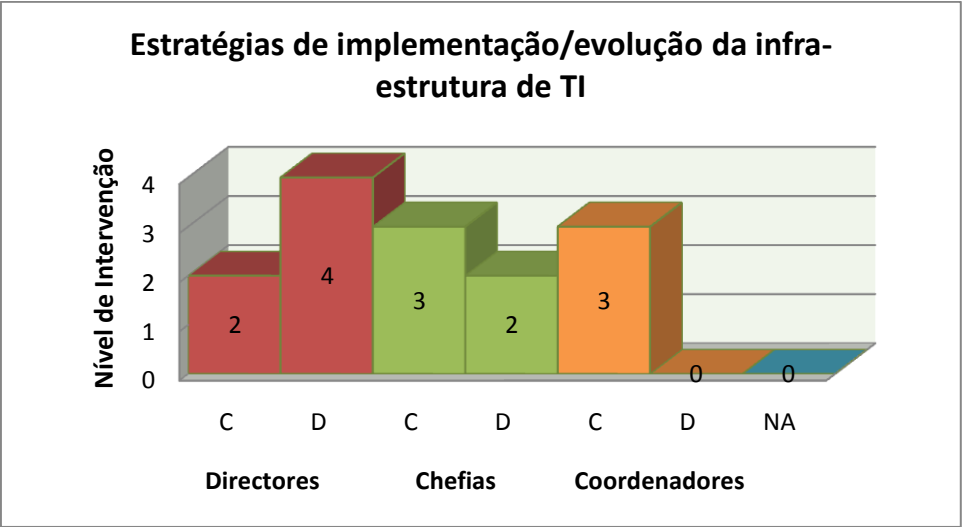
a)



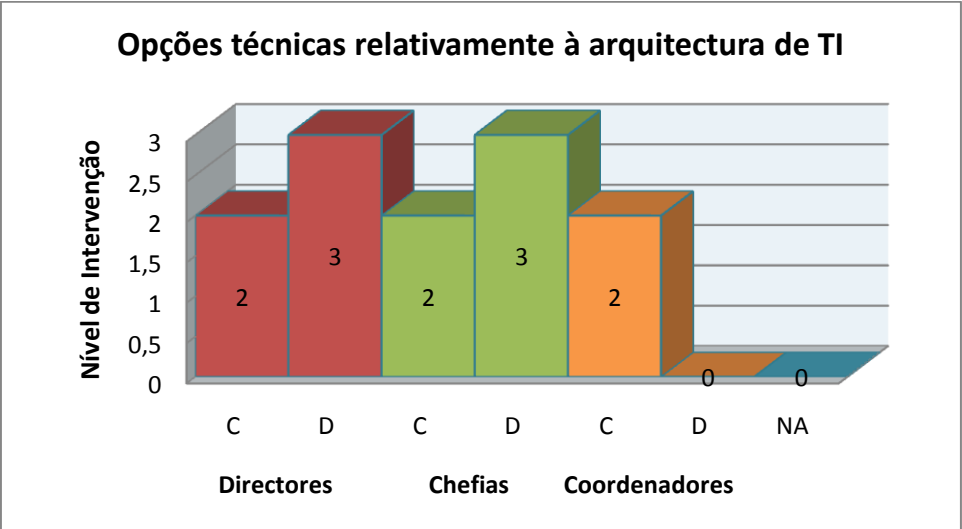
b)



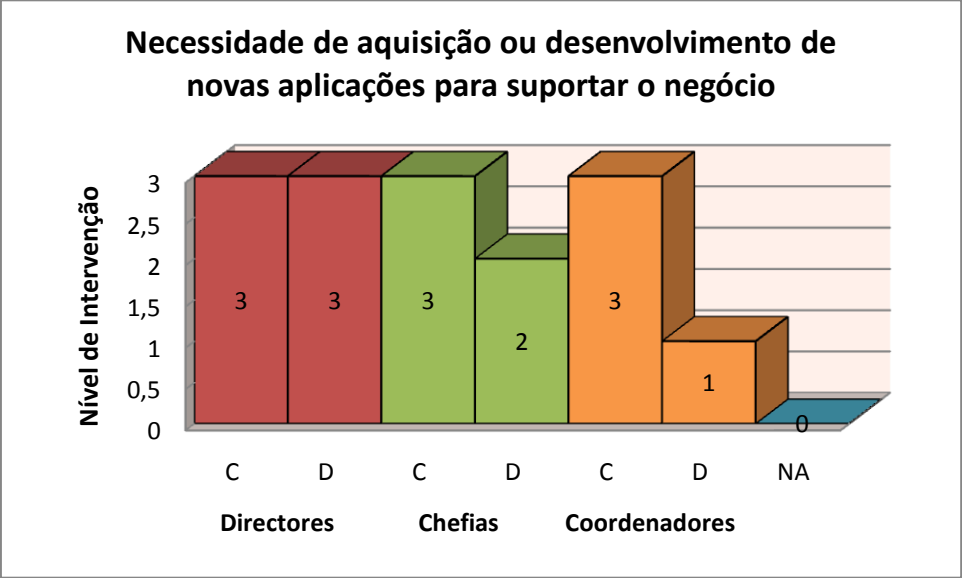
c)



d)



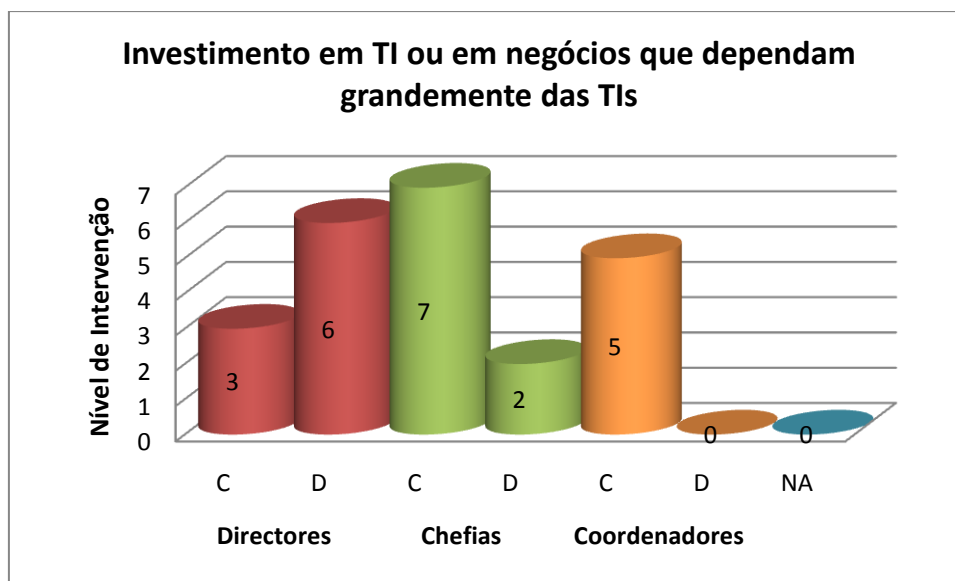
e)



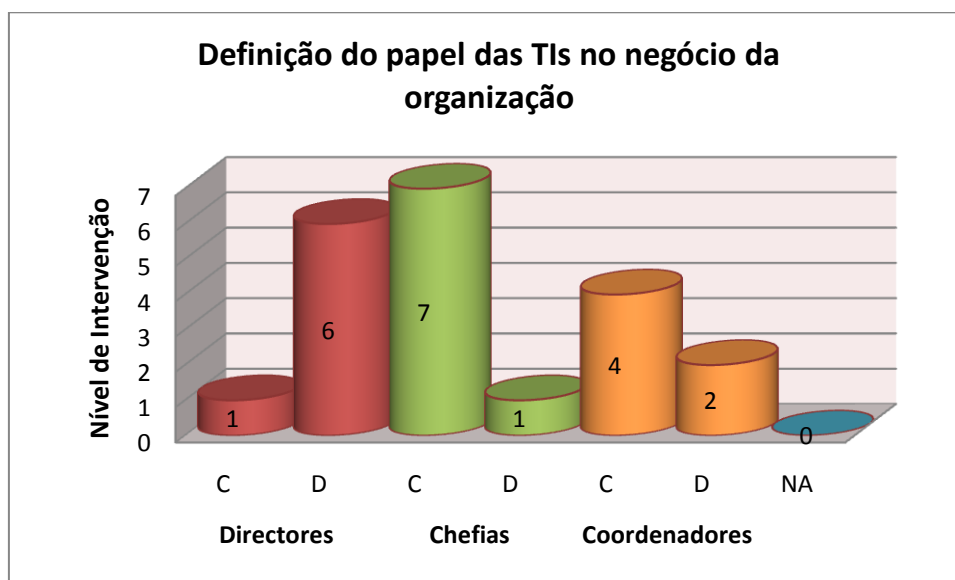
➤ **Médias Empresas** (< 250 colaboradores)

1. Nível de intervenção (Contribui ou Decide) de cada um dos seguintes actores relativamente a:

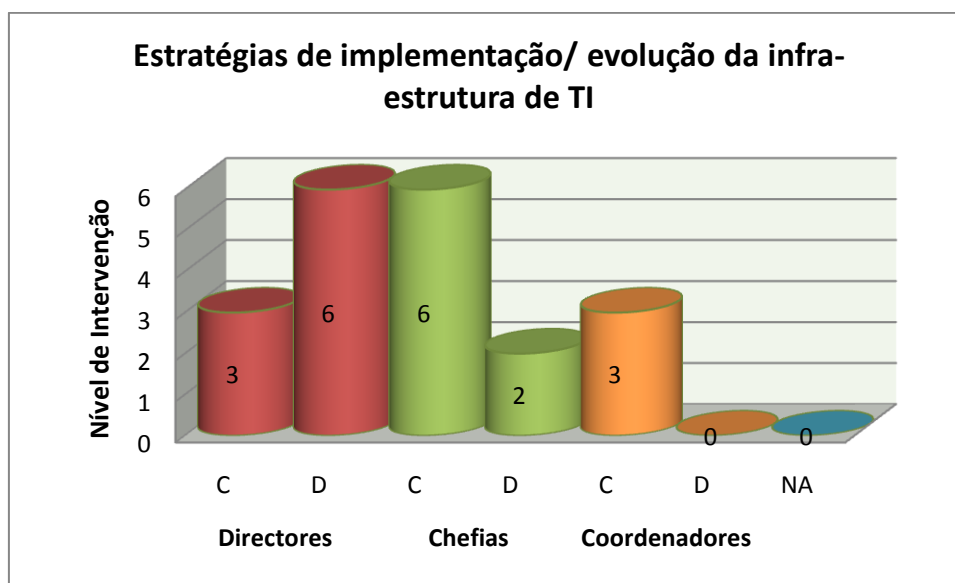
a)



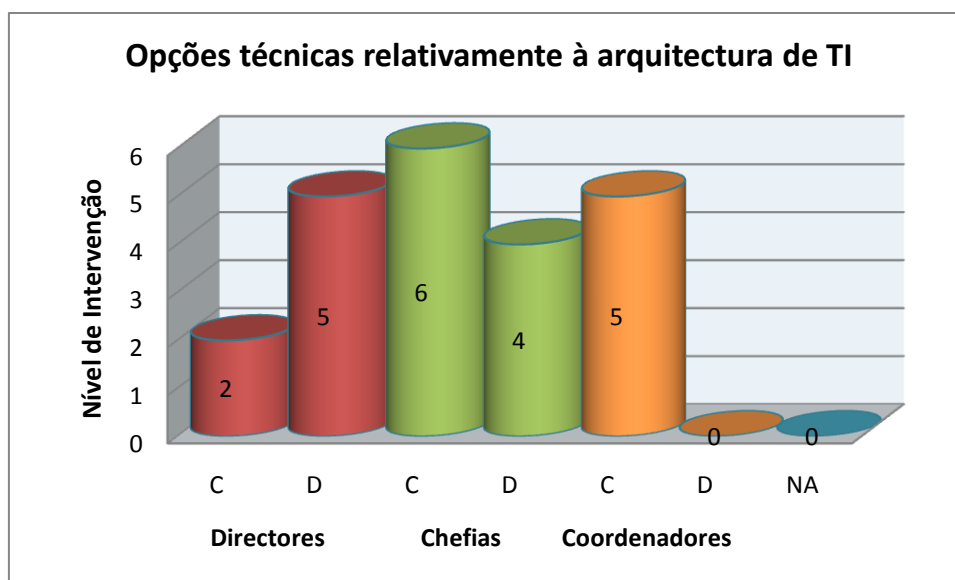
b)



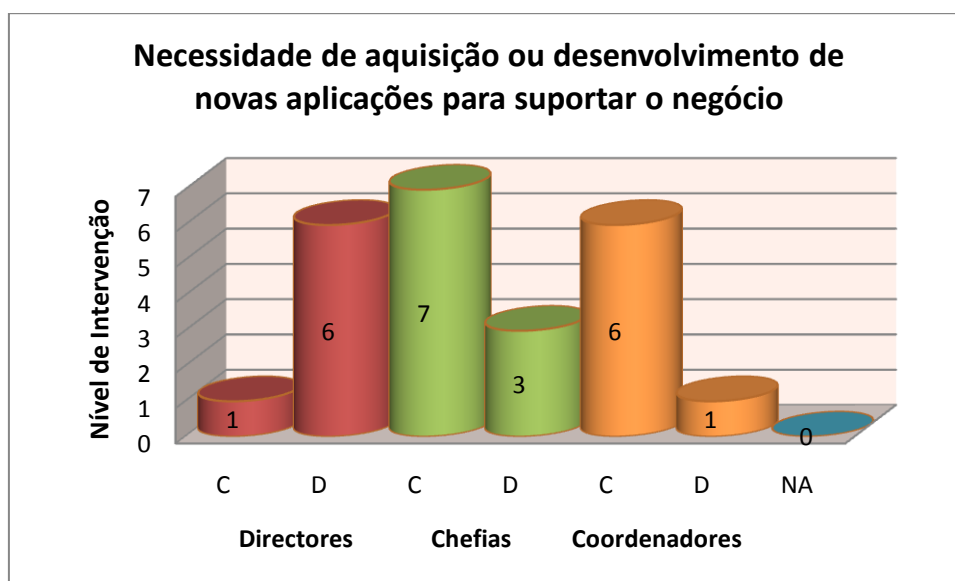
c)



d)



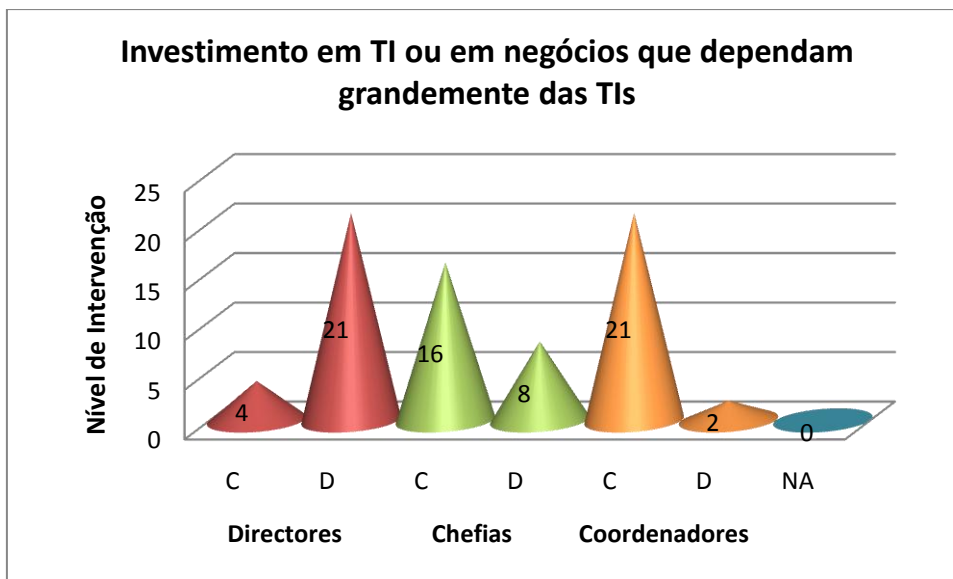
e)



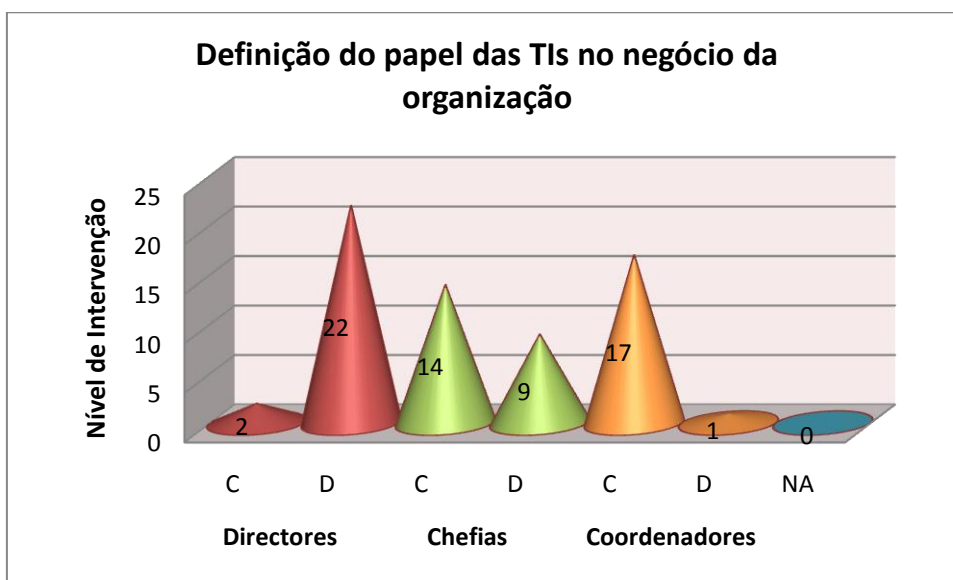
➤ **Grandes Empresas** (> 250 colaboradores)

1. Nível de intervenção (Contribui ou Decide) de cada um dos seguintes actores relativamente a:

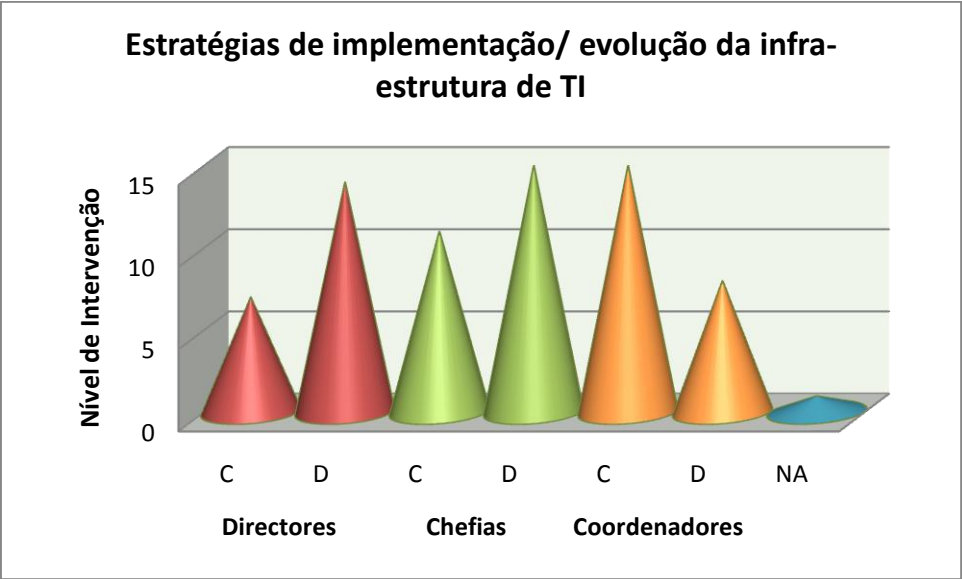
a)



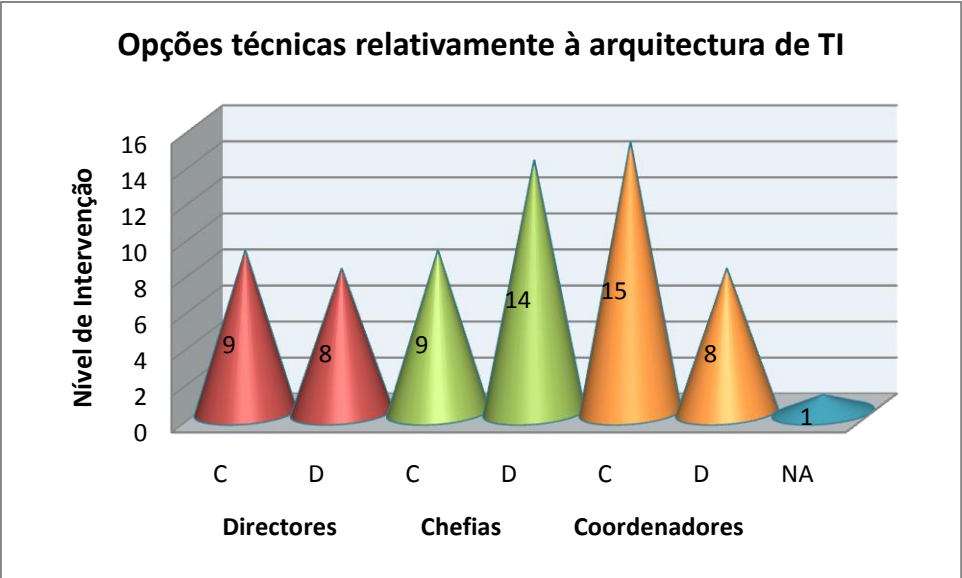
b)



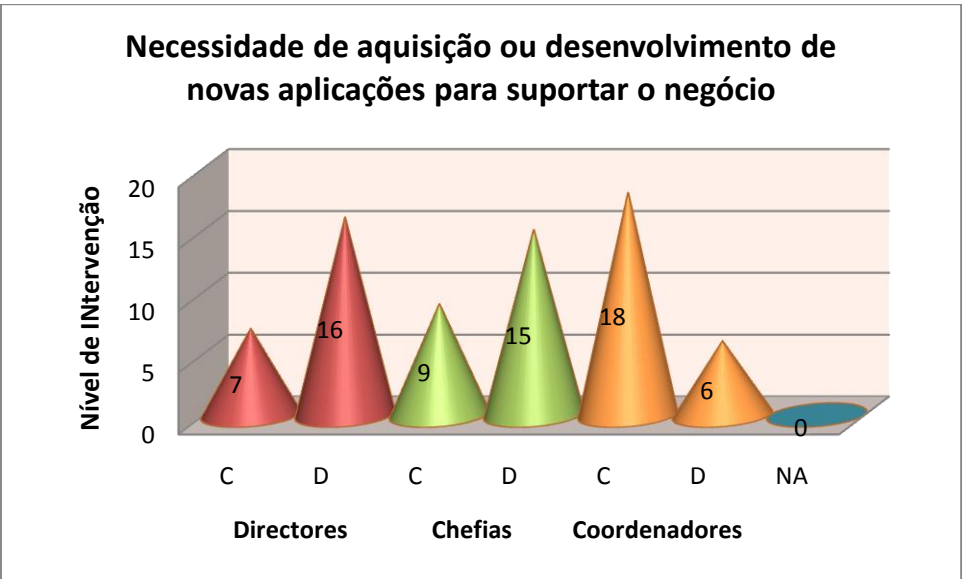
c)



d)



e)

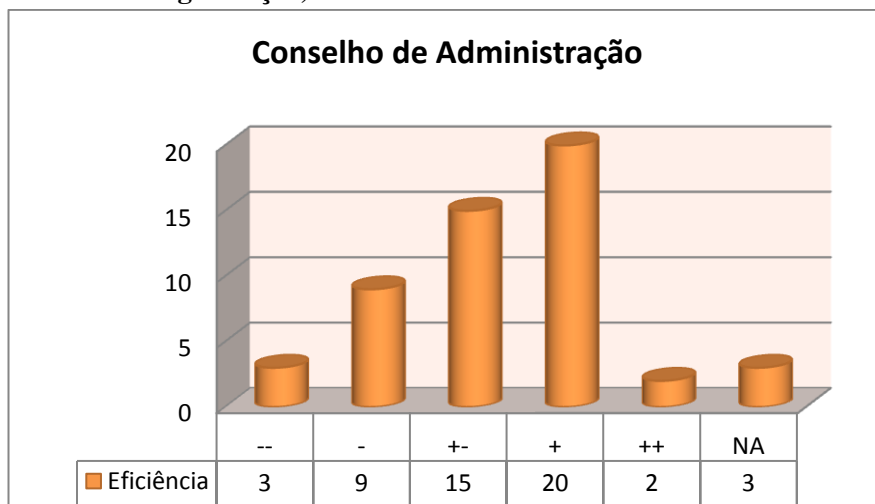


2. Órgãos com intervenção na governança de TI

Nas organizações existem, com frequência, uma variedade de órgãos com maior ou menor intervenção na governança de TI.

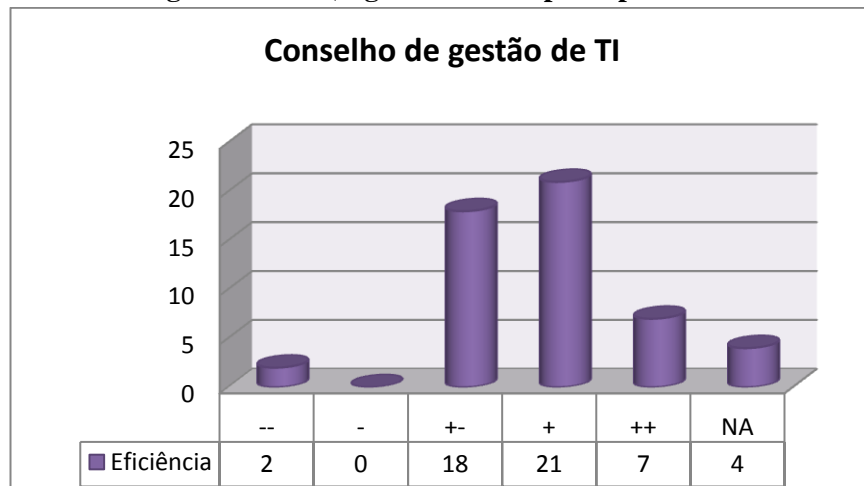
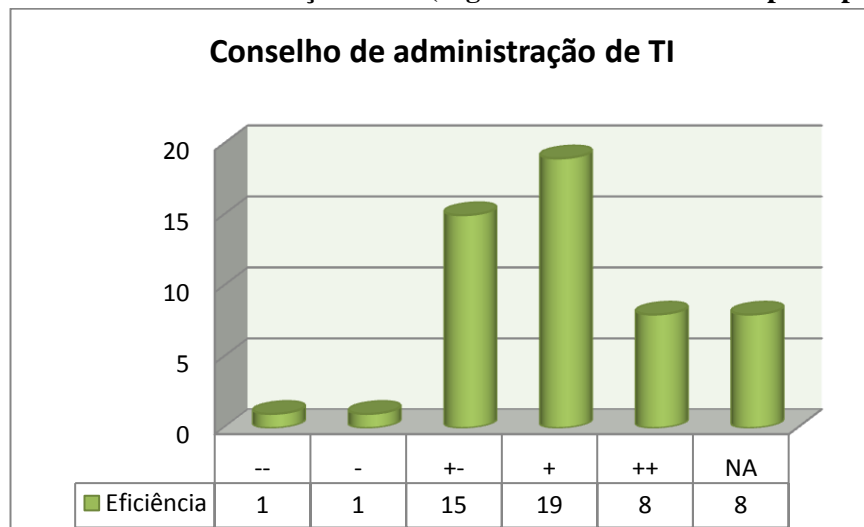
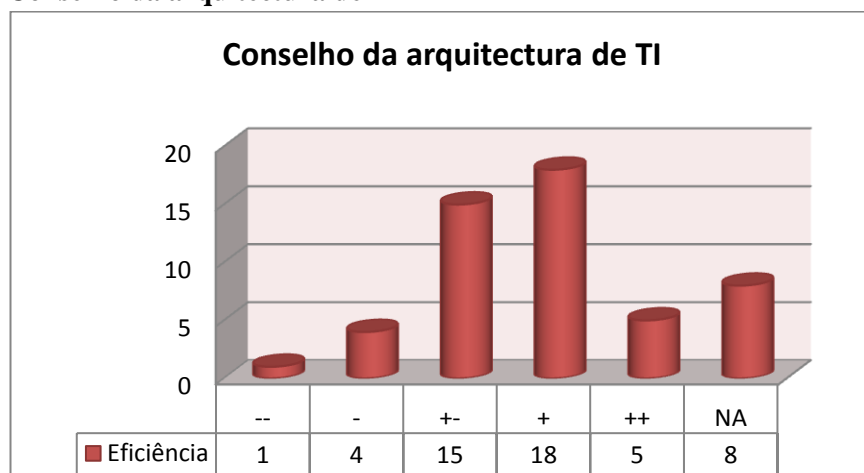
Avaliação da eficiência desses órgãos para potenciar o valor do negócio das TIs nas organizações:

2.1. Conselho de Administração (constituído maioritariamente pelos directores dos vários sectores da organização)

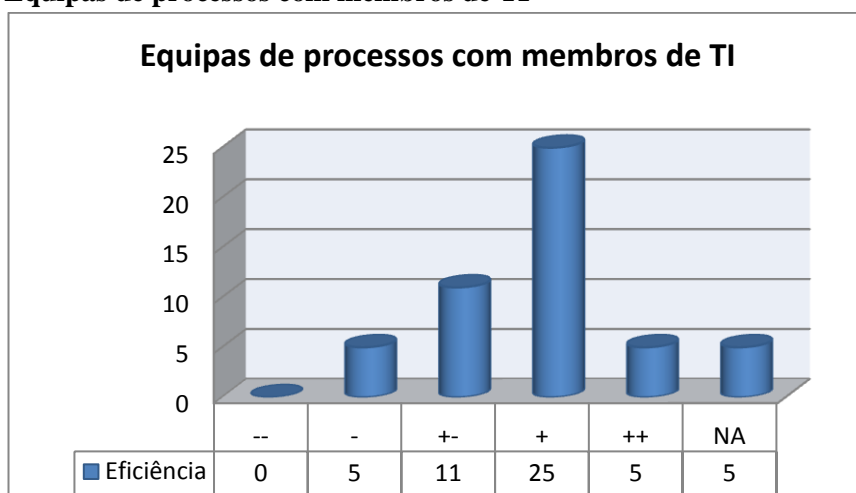


2.2. Comissão de aprovação do orçamento

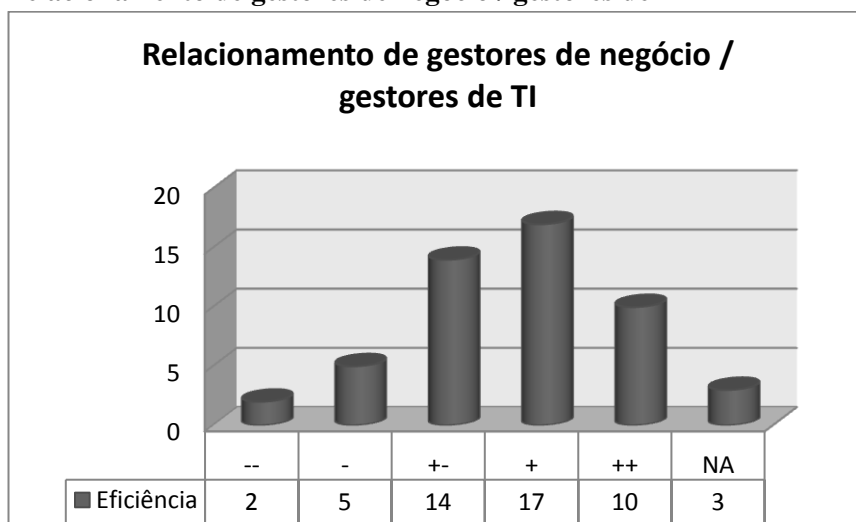


2.3. Conselho de gestão de TI (órgão misto composto por executivos de TI e de negócio)**2.4. Conselho de administração de TI (órgão exclusivamente composto por executivos de TI)****2.5. Conselho da arquitectura de TI**

2.6. Equipas de processos com membros de TI



2.7. Relacionamento de gestores de negócio / gestores de TI

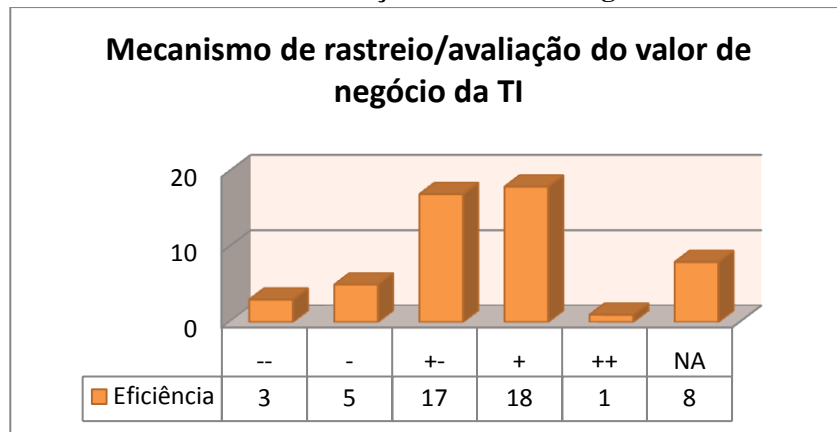


3. Mecanismos de Governança de TI

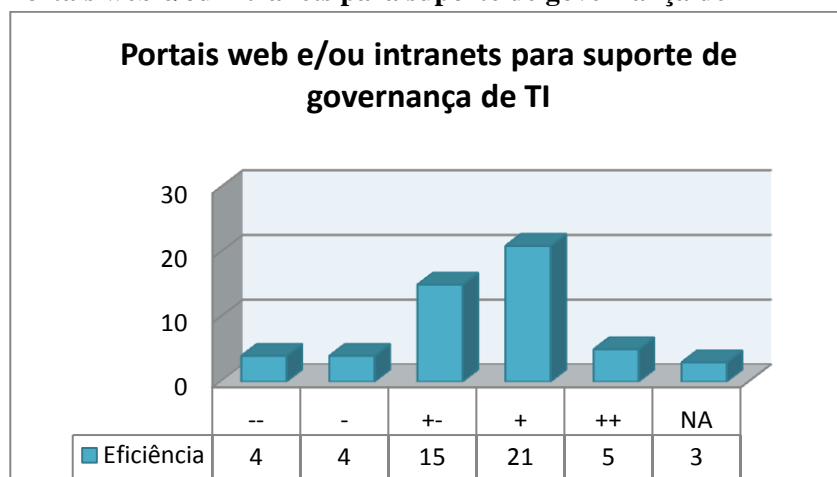
Nas organizações existem, com frequência, uma variedade de mecanismos de governança de TI.

Avaliação da eficiência desses mecanismos para potenciar o valor do negócio das TIs na sua organização:

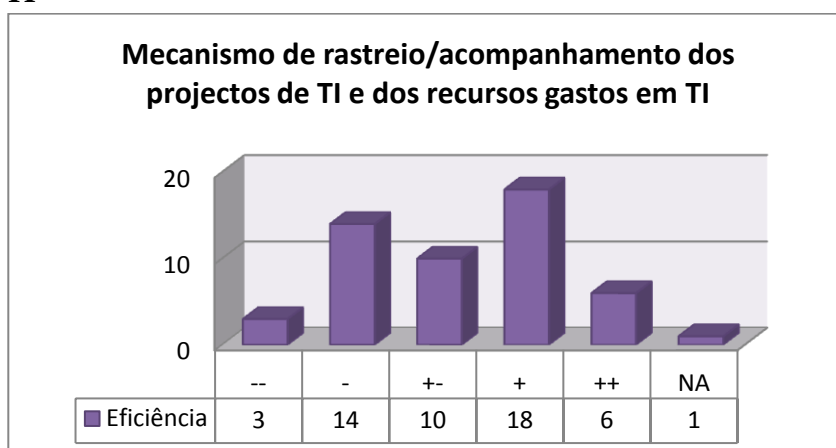
3.1 Mecanismo de rastreio/avaliação do valor de negócio da TI



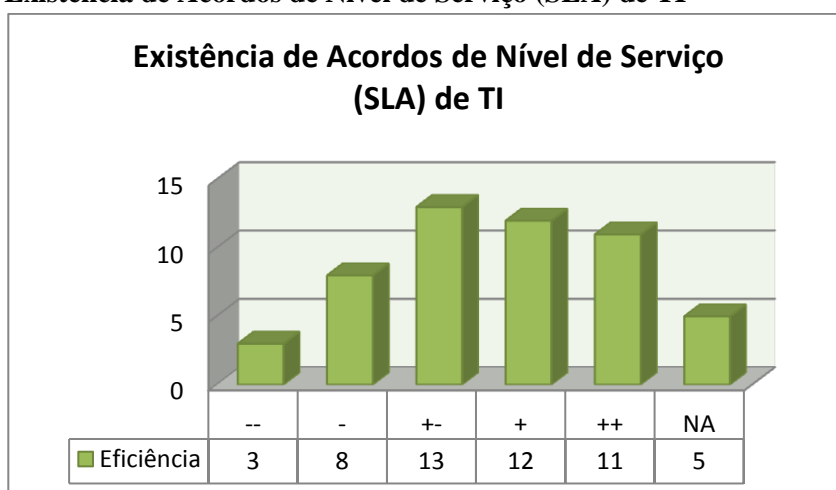
3.2 Portais web e/ou intranets para suporte de governança de TI



3.3 Mecanismo de rastreio/acompanhamento dos projectos de TI e dos recursos gastos em TI



3.4 Existência de Acordos de Nível de Serviço (SLA) de TI



3.5 Existência de Acordos internos de recuperação de custos de TI

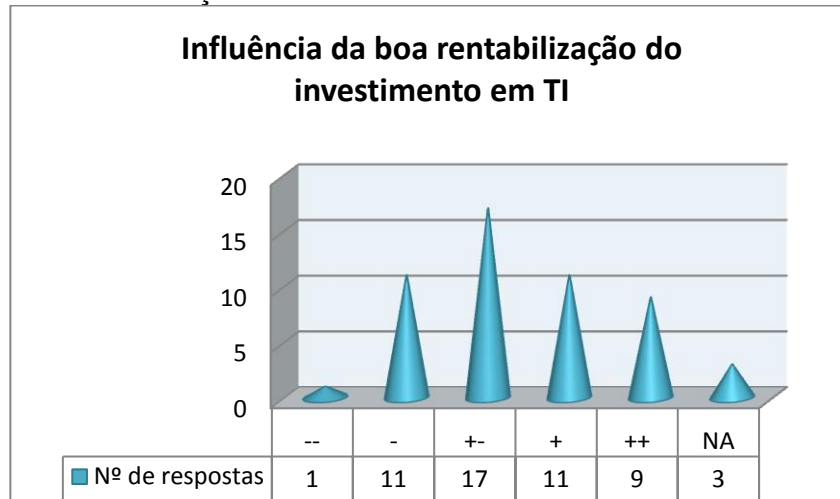


B. Desempenho da governança de TI

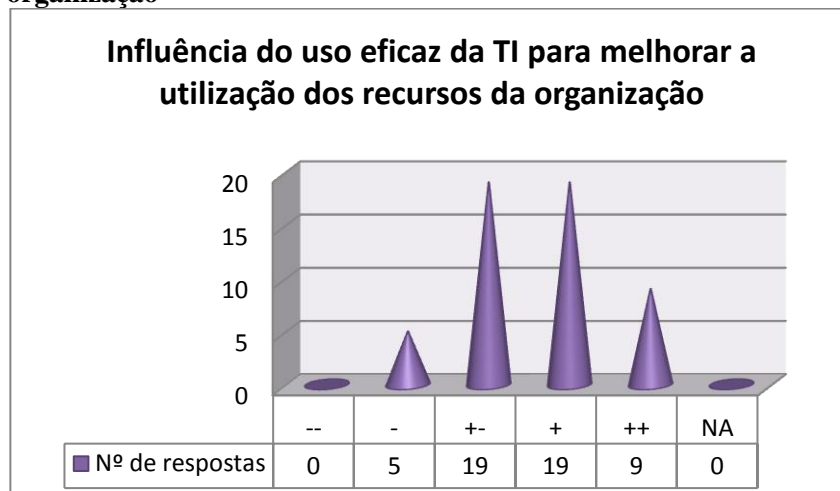
1. Objectivos de governança de TI

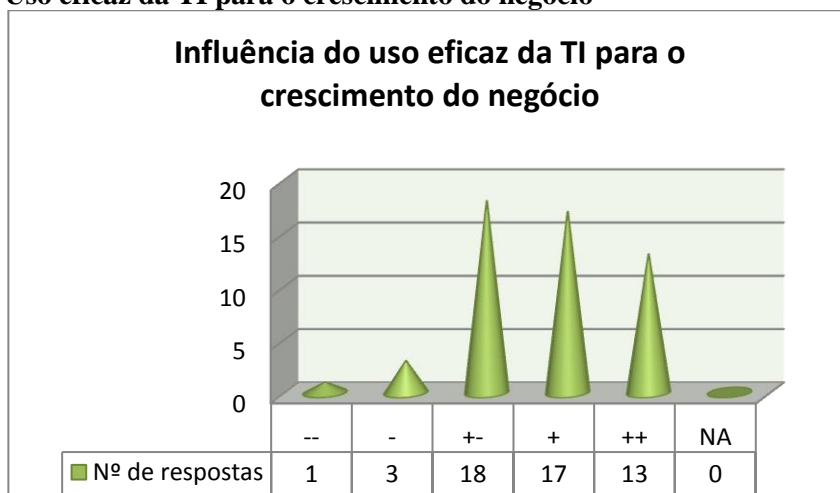
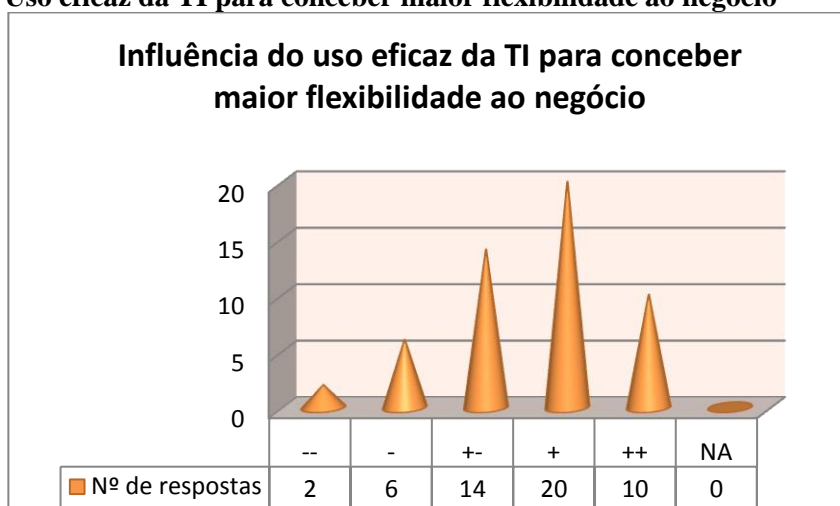
Qual tem sido (no passado) a influência da governança de TI das organizações relativamente aos seguintes aspectos.

1.1 Boa rentabilização do investimento em TI



1.2 Uso eficaz da TI para melhorar a utilização dos recursos (humanos e materiais) da organização

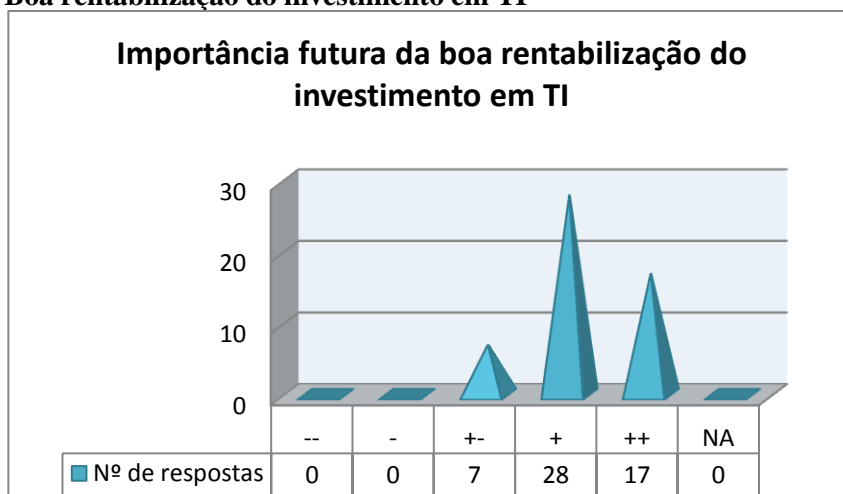


1.3 Uso eficaz da TI para o crescimento do negócio**1.4 Uso eficaz da TI para conceber maior flexibilidade ao negócio**

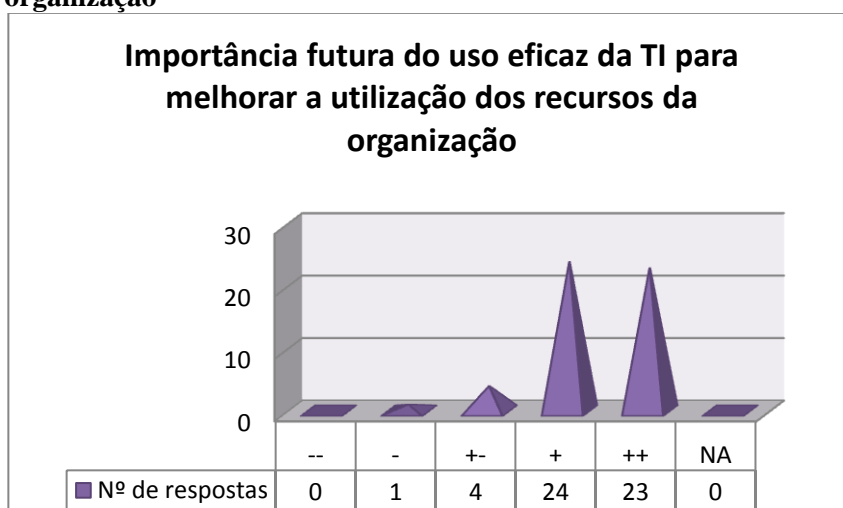
2. Resultados de governança de TI

Qual é a importância que a governança de TI das organizações deveria dar (no futuro) aos seguintes aspectos:

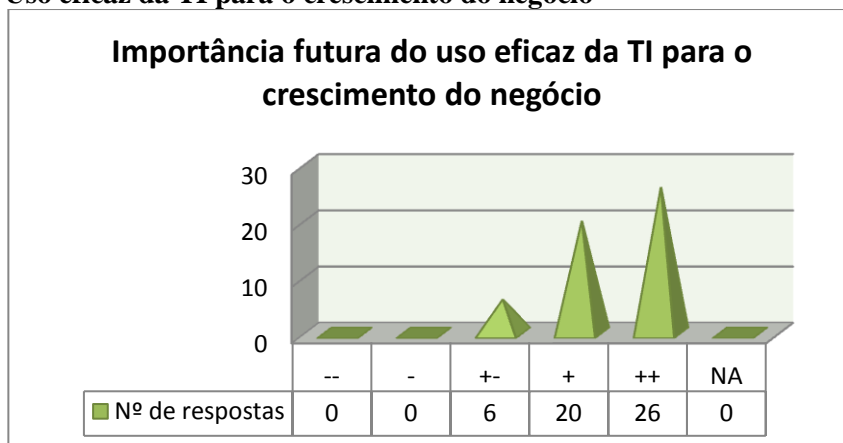
2.1 Boa rentabilização do investimento em TI

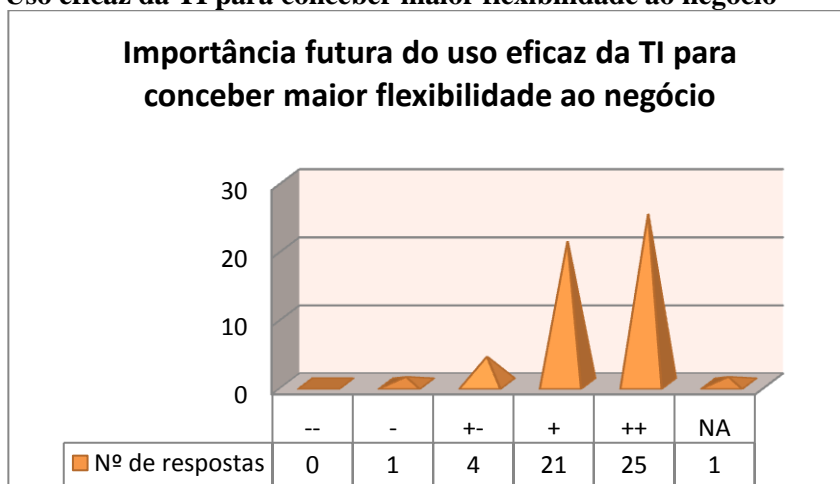


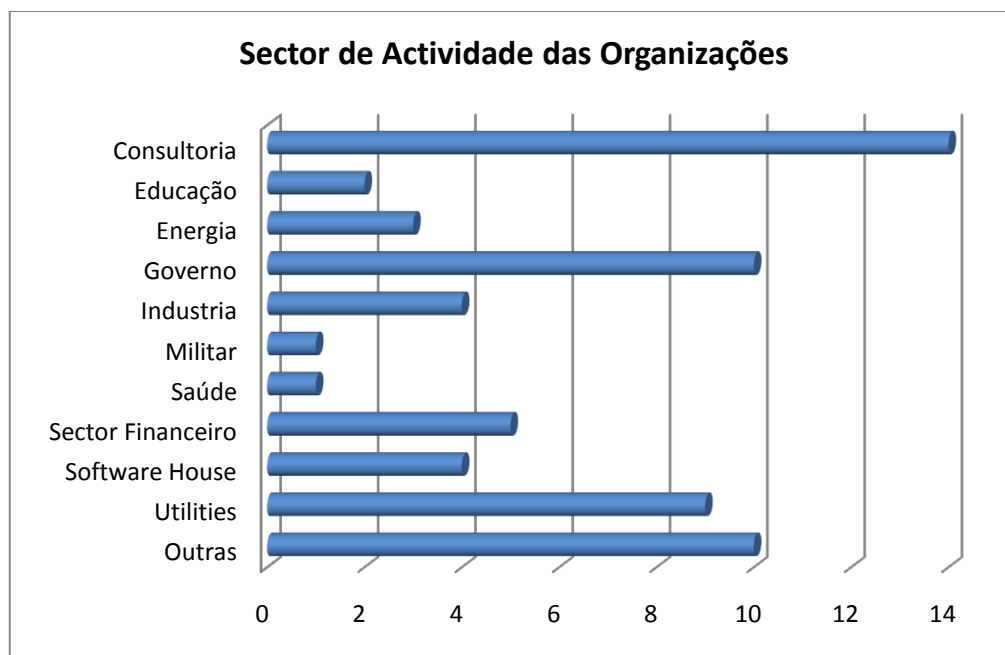
2.2 Uso eficaz da TI para melhorar a utilização dos recursos (humanos e materiais) da organização



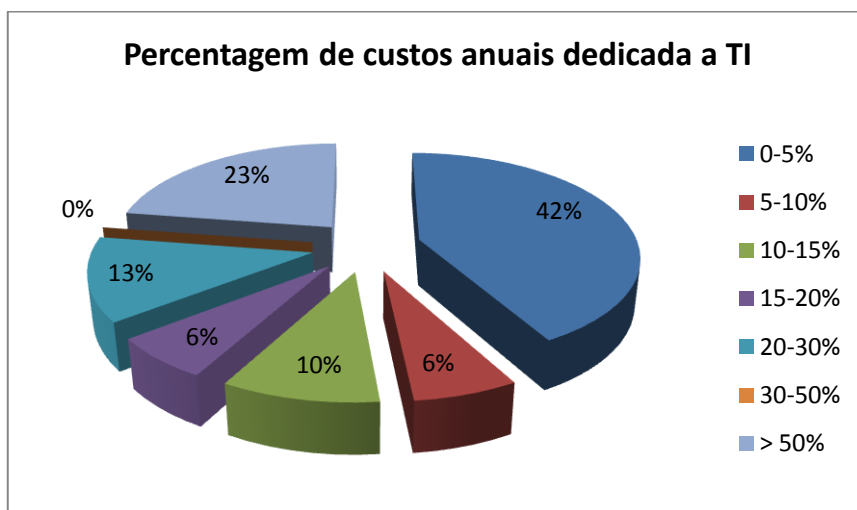
2.3 Uso eficaz da TI para o crescimento do negócio



2.4 Uso eficaz da TI para conceber maior flexibilidade ao negócio

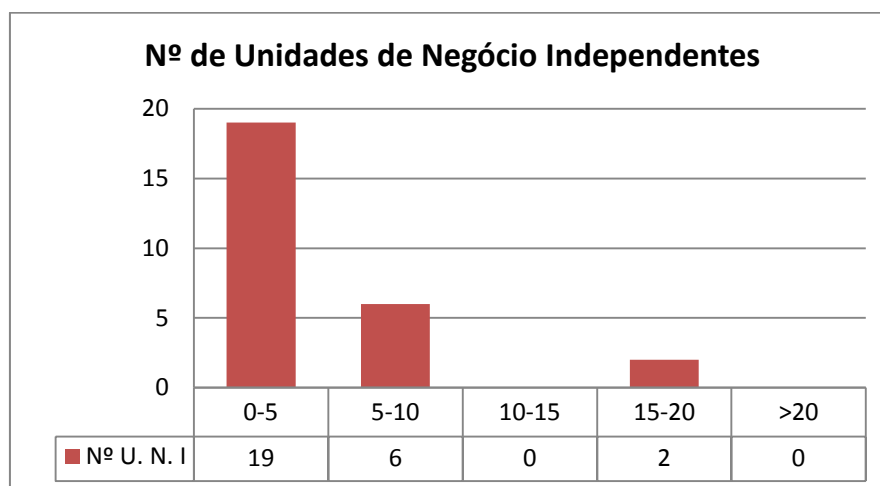
C. Contexto Organizacional**1. Sector de actividade das organizações:****2. Tipos de organização/sector:**

3. Percentagem de custos anuais dedicada à TI

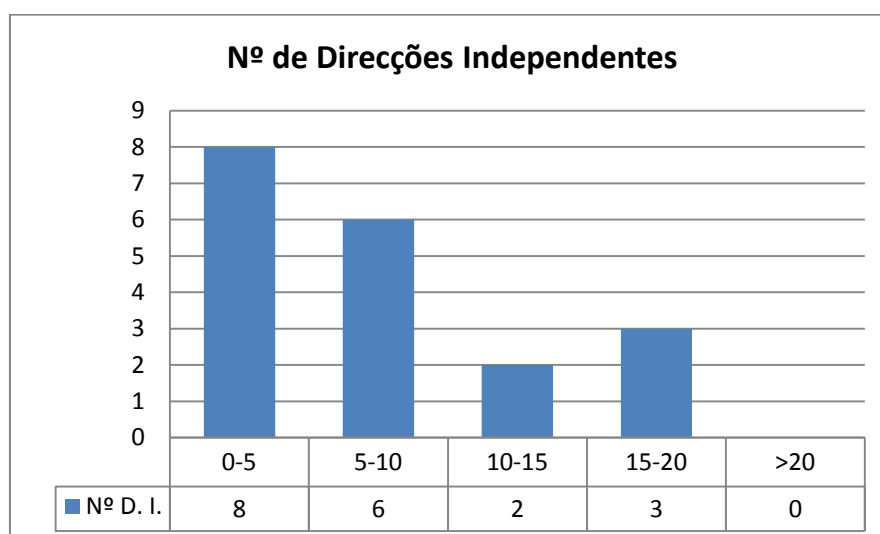


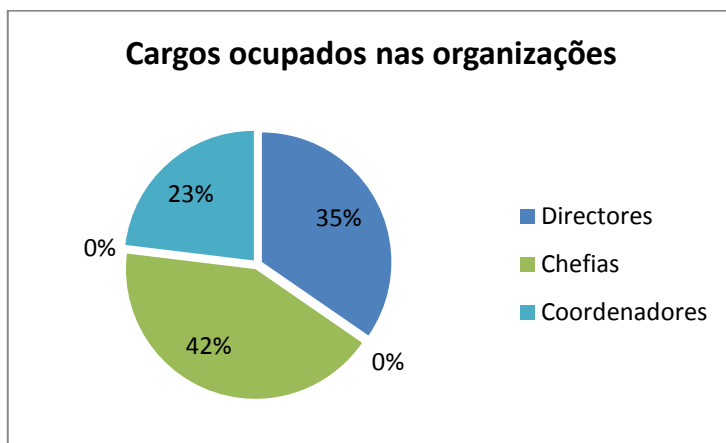
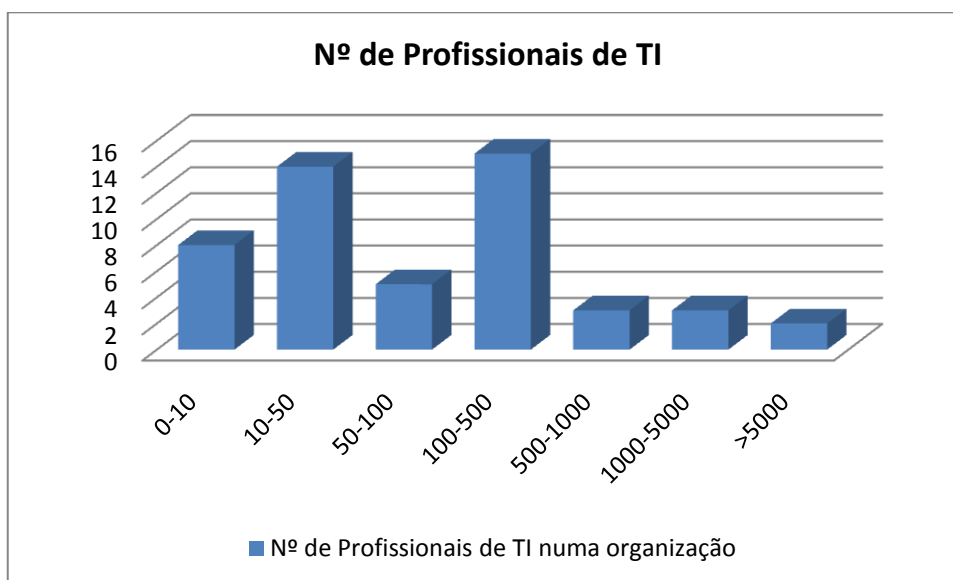
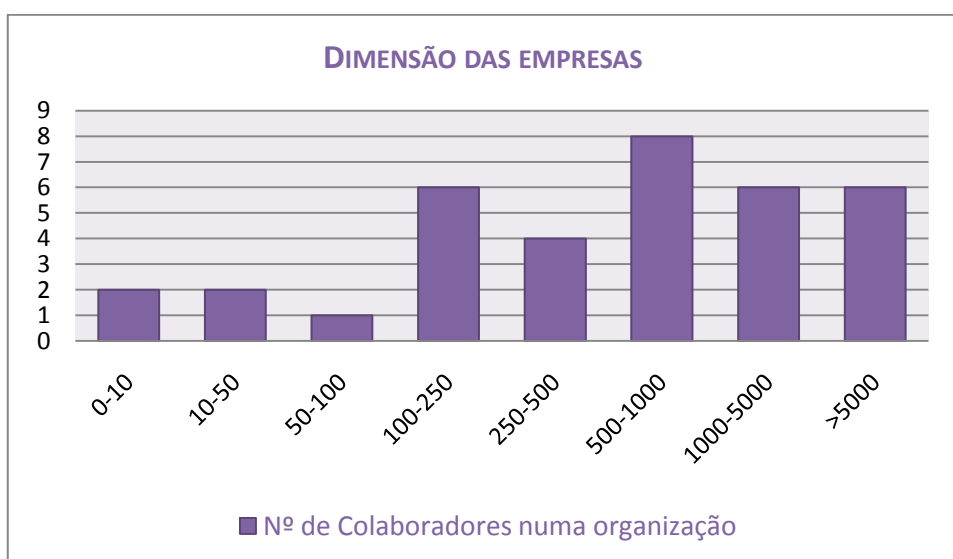
4. Dependendo do tipo de estrutura das organizações, indicaram:

a. Número de unidades de negócio independentes



b. Número de direcções independentes



5. Cargos que os colaboradores ocupam nas organizações**6. Total aproximado de profissionais de TI nas organizações****7. Total aproximado de colaboradores das organizações**

c) Interpretação dos resultados do inquérito sobre governança de TI

1. Perfil de Governança de TI (com valor médio dos 52 inquéritos respondidos)

Resultados obtidos através das respostas à pergunta 1 do Grupo A

Domain Style	IT Principles	IT Architecture	IT Infrastructure	Business Application Needs	IT Investment and Prioritization
Business Monarchy	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
IT Monarchy		<input checked="" type="checkbox"/>			
Feudal					
Federal					
Duopoly				<input checked="" type="checkbox"/>	
Anarchy					

Domain Style	IT Principles	IT Architecture	IT Infrastructure	Business Application Needs	IT Investment and Prioritization
Business Monarchy	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
IT Monarchy					
Feudal					
Federal					
Duopoly				<input checked="" type="checkbox"/>	
Anarchy					

Temos dois tipos de perfil de Governança de TI, porque o domínio de decisão “IT Architecture” não ficou claro, ou seja, tanto podem ser os Directores a decidirem (Business Monarchy) como as chefias a decidirem (IT Monarchy).

d)	Directores	Contribui	23
		Decide	22
	Chefias	Contribui	25
		Decide	30
	Coordenadores	Contribui	35
		Decide	11
	NA	NA	8

- Pontos em comum com os “Top Three Performers”

1º Perfil			2º Perfil		
	Pontos em comum	Percentagem		Pontos em comum	Percentagem
1	1	1/5 -> 20%	1	0	0/5 -> 0%
2	3	3/5 -> 60%	2	2	2/5 -> 40%
3	3	3/5 -> 60%	3	4	4/5 -> 80%

O perfil de Governança de TI das empresas portuguesas situa-se entre [60-80]%

2. Desempenho de Governança de TI (com valor médio dos 52 inquéritos respondidos)

Resultados obtidos através das respostas à pergunta 1e 2 do Grupo B

Desempenho da governança

$$= \frac{\sum_{n=1}^4 (\text{importância do resultado } \{i. e \text{ Questão } 1\} * \text{influência da governança de TI } \{i. e \text{ Questão } 2\}) * 100}{\sum_{n=1}^4 (5(\text{importância do resultado}))}$$

Importância/influência da Governança de TI	Média Questão 1	Média Questão 2
Boa rentabilização do investimento em TI	3,326531	4,192308
Uso eficaz da TI para melhorar a utilização dos recursos	3,615385	4,326923
Uso eficaz da TI para o crescimento do negócio	3,730769	4,384615
Uso eficaz da TI para conceber maior flexibilidade ao negócio	3,576923	4,372549

$$\text{Desempenho da governança} = \frac{61,58759066}{71,24803768} * 100 = 86,4411044 \approx 86 \%$$

➤ **Pequenas Empresas** (< 50 colaboradores)

1. **Perfil de Governança de TI para pequenas empresas** (com valor médio dos 4 inquéritos respondidos)

Domain Style	IT Principles	IT Architecture	IT Infrastructure	Business Application Needs	IT Investment and Prioritization
Business Monarchy	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
IT Monarchy					
Feudal					
Federal					
Duopoly		<input checked="" type="checkbox"/>			
Anarchy					

- Pontos em comum com os “Top Three Performers”

	Pontos em comum	Percentagem
1	0	0/5 -> 0%
2	1	1/5 -> 20%
3	3	3/5 -> 60%

O perfil de Governança de TI das pequenas empresas portuguesas situa-se nos 60%. Não é um valor fidedigno uma vez que só tínhamos uma amostra de 4 inquéritos.

2. **Desempenho de Governança de TI** (com valor médio dos 4 inquéritos respondidos)

Resultados obtidos através das respostas à pergunta 1e 2 do Grupo B

Desemepenho da governança

$$= \frac{\sum_{n=1}^4 (\text{importância do resultado } \{i. e \text{ Questão } 1\} * \text{influência da governança de TI } \{i. e \text{ Questão } 2\}) * 100}{\sum_{n=1}^4 (5(\text{importância do resultado}))}$$

Importância/influência da Governança de TI	Média Questão 1	Média Questão 2
Boa rentabilização do investimento em TI	0,4	0,485714286
Uso eficaz da TI para melhorar a utilização dos recursos	0,428571429	0,542857143
Uso eficaz da TI para o crescimento do negócio	0,514285714	0,542857143
Uso eficaz da TI para conceber maior flexibilidade ao negócio	0,485714286	0,542857143

Desempenho da governança = 10,60714286 ≈ 11 %

➤ **Médias Empresas** (< 250 colaboradores)

1. **Perfil de Governança de TI para médias empresas** (com valor médio dos 7 inquéritos respondidos)

Domain Style	IT Principles	IT Architecture	IT Infrastructure	Business Application Needs	IT Investment and Prioritization
Business Monarchy	✗	✗	✗	✗	✗
IT Monarchy					
Feudal					
Federal					
Duopoly					
Anarchy					

- Pontos em comum com os “Top Three Performers”

	Pontos em comum	Porcentagem
1	0	0/5 -> 0%
2	1	1/5 -> 20%
3	4	4/5 -> 80%

O perfil de Governança de TI das médias empresas portuguesas situa-se nos 80%.

Não é um valor fidedigno, uma vez que, só tínhamos uma amostra de 7 inquéritos.

2. **Desempenho de Governança de TI** (com valor médio dos 7 inquéritos respondidos)

Resultados obtidos através das respostas à pergunta 1e 2 do Grupo B

Desempenho da governança

$$= \frac{\sum_{n=1}^4 (\text{importância do resultado } \{i. e \text{ Questão } 1\} * \text{influência da governança de TI } \{i. e \text{ Questão } 2\}) * 100}{\sum_{n=1}^4 (5(\text{importância do resultado}))}$$

Importância/influência da Governança de TI	Média Questão 1	Média Questão 2
Boa rentabilização do investimento em TI	0,617647059	0,882352941
Uso eficaz da TI para melhorar a utilização dos recursos	0,676470588	0,941176471
Uso eficaz da TI para o crescimento do negócio	0,647058824	0,911764706
Uso eficaz da TI para conceber maior flexibilidade ao negócio	0,705882353	0,941176471

Desempenho da governança = 18,40522876 ≈ 18 %

➤ **Grandes Empresas** (> 250 colaboradores)

1. Perfil de Governança de TI para grandes empresas (com valor médio dos 23 inquéritos respondidos)

Domain Style	IT Principles	IT Architecture	IT Infrastructure	Business Application Needs	IT Investment and Prioritization
Business Monarchy	✗				✗
IT Monarchy		✗			
Feudal					
Federal					
Duopoly			✗	✗	
Anarchy					

- Pontos em comum com os “Top Three Performers”

	Pontos em comum	Percentagem
1	1	1/5 -> 20%
2	3	3/5 -> 60%
3	2	2/5 -> 40%

O perfil de Governança de TI das grandes empresas portuguesas situa-se nos 60%.

2. Desempenho de Governança de TI (com valor médio dos 7 inquéritos respondidos)

Resultados obtidos através das respostas à pergunta 1e 2 do Grupo B

Desempenho da governança

$$= \frac{\sum_{n=1}^4 (\text{importância do resultado } \{i.e \text{ Questão } 1\} * \text{influência da governança de TI } \{i.e \text{ Questão } 2\}) * 100}{\sum_{n=1}^4 (5(\text{importância do resultado}))}$$

Importância/influência da Governança de TI	Média Questão 1	Média Questão 2
Boa rentabilização do investimento em TI	2,090909091	2,939393939
Uso eficaz da TI para melhorar a utilização dos recursos	2,666666667	2,96969697
Uso eficaz da TI para o crescimento do negócio	2,878787879	3,090909091
Uso eficaz da TI para conceber maior flexibilidade ao negócio	2,666666667	2,909090909

Desempenho da governança = 59,63458111 ≈ 60 %

Em Conclusão:**1. Perfil de Governança de TI** (médias dos 52 inquéritos)

Domain \ Style	IT Principles	IT Architecture	IT Infrastructure	Business Application Needs	IT Investment and Prioritization
Business Monarchy					
IT Monarchy					
Feudal					
Federal					
Duopoly					
Anarchy					

	80 %
	60 %

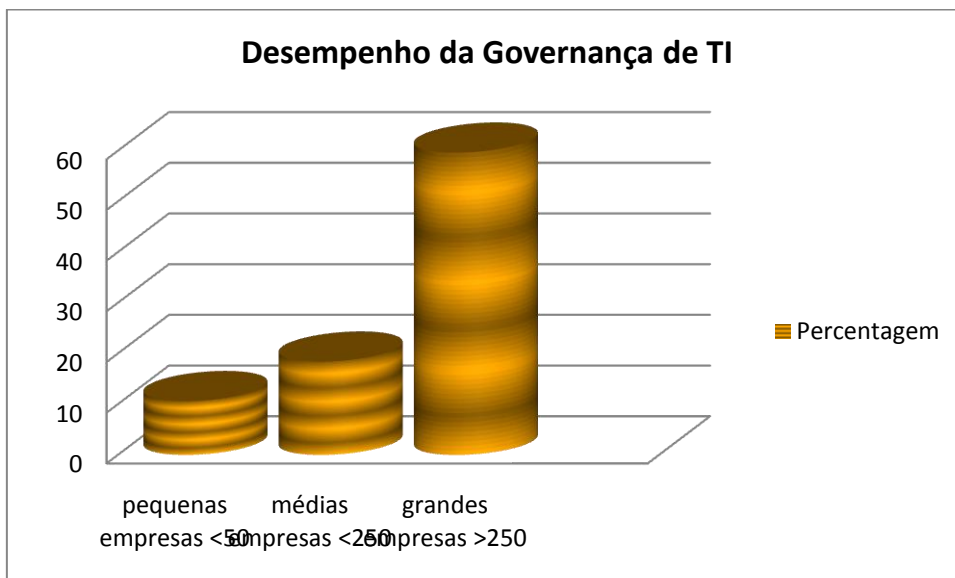
Desempenho de Governança de TI (médias dos 52 inquéritos)

Desempenho da governança ≈ 86 %

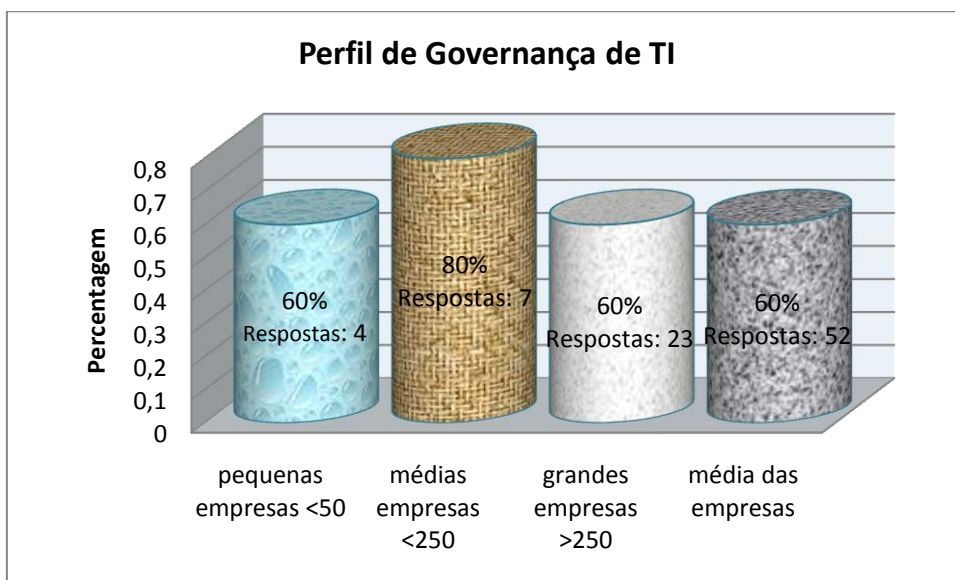
2. Perfil de Governança de TI (Pequenas, Médias e Grandes Empresas)

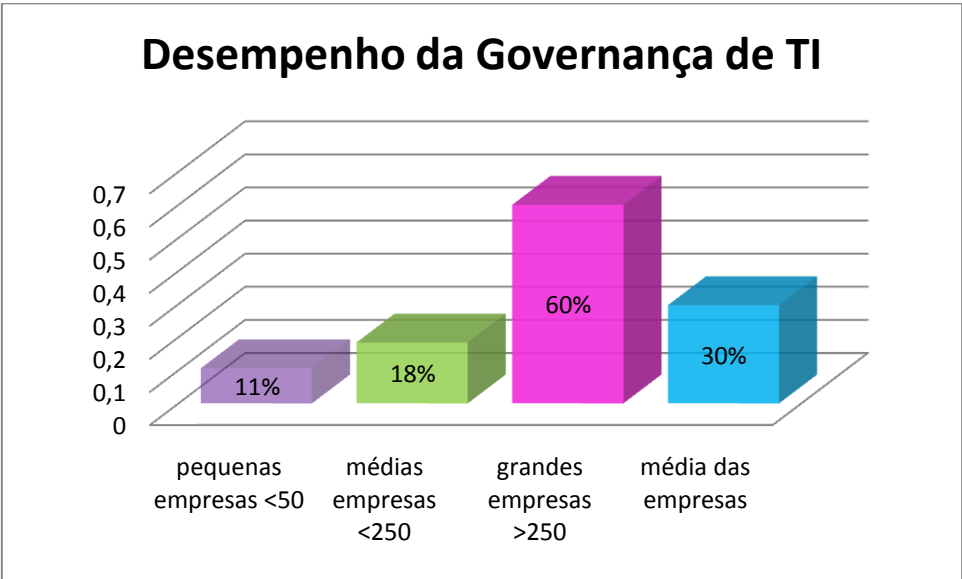
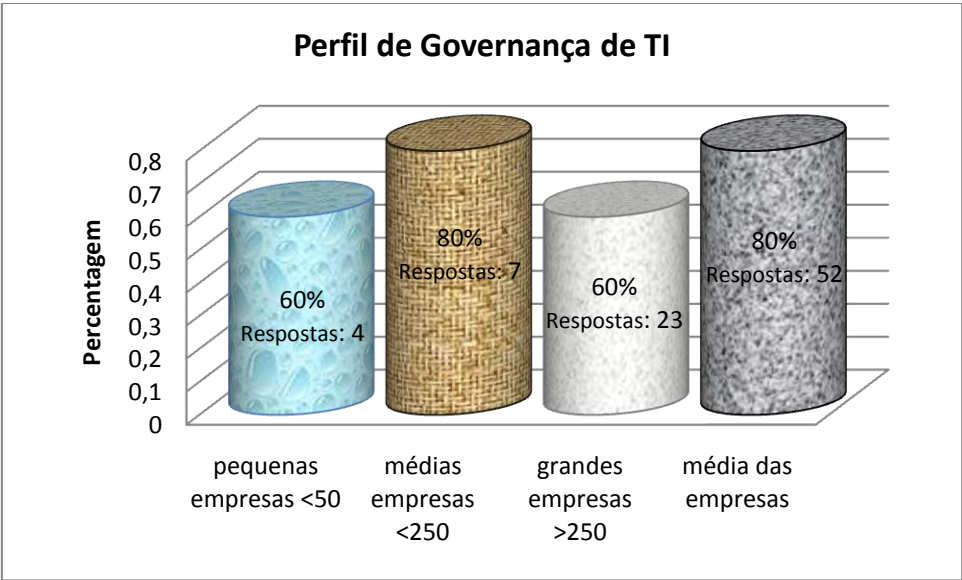
Domain \ Style	IT Principles	IT Architecture	IT Infrastructure	Business Application Needs	IT Investment and Prioritization
Business Monarchy					
IT Monarchy					
Feudal					
Federal					
Duopoly					
Anarchy					

	60 %
	80 %
	60 %

Desempenho de Governança de TI (Pequenas, Médias e Grandes Empresas)**Quadro Síntese**

	Perfil	Desempenho
Média	[60 - 80] %	86 %
Pequenas Empresas	60 %	11 %
Médias Empresas	80 %	18 %
Grandes Empresas	60 %	60 %





[Esta página foi intencionalmente deixada em branco]

Anexo B

Metamodelo BPMN e restrições OCL

Conteúdo

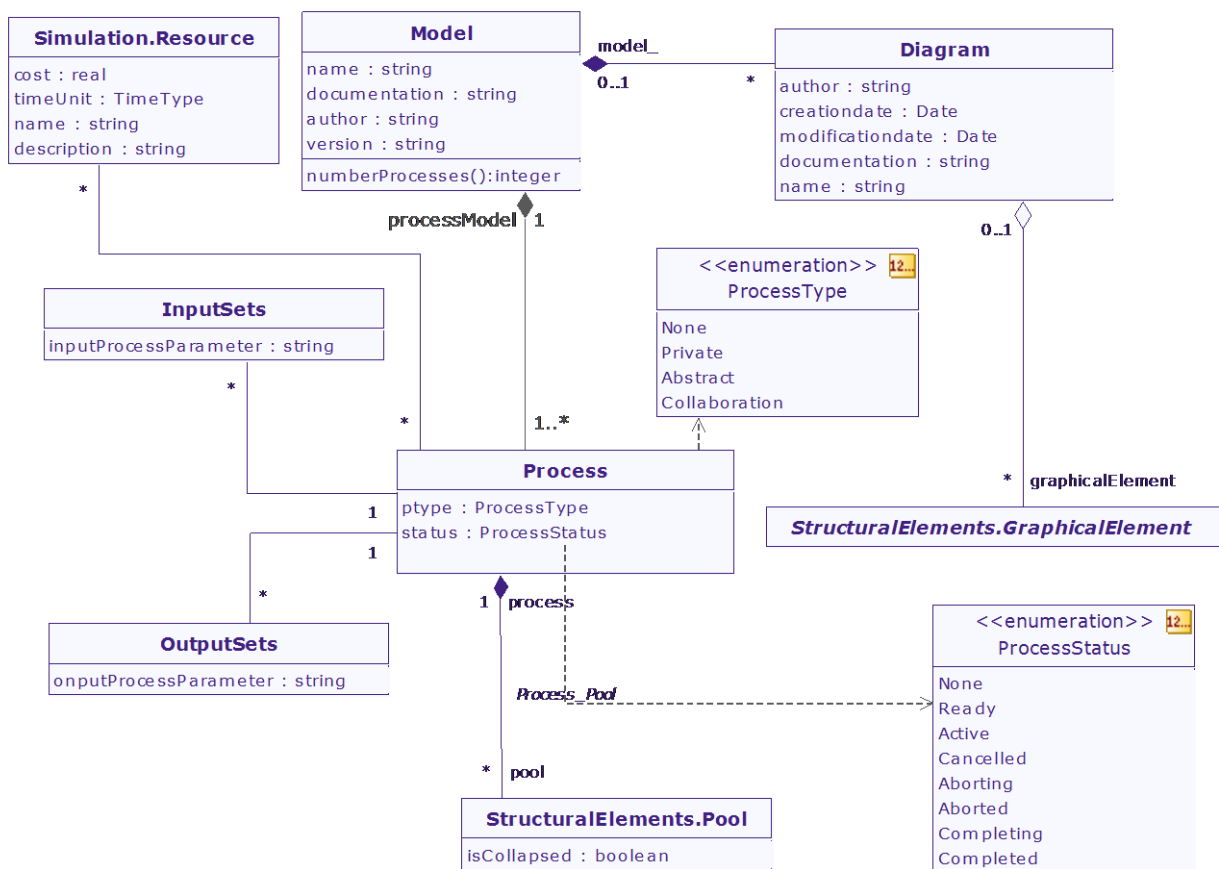
B. META MODELO BPMN E RESTRIÇÕES OCL.....	168
A) META MODELO BPMN.....	168
B) MÉTRICAS OCL	177

Este anexo ilustra o metamodelo BPMN e as métricas OCL.

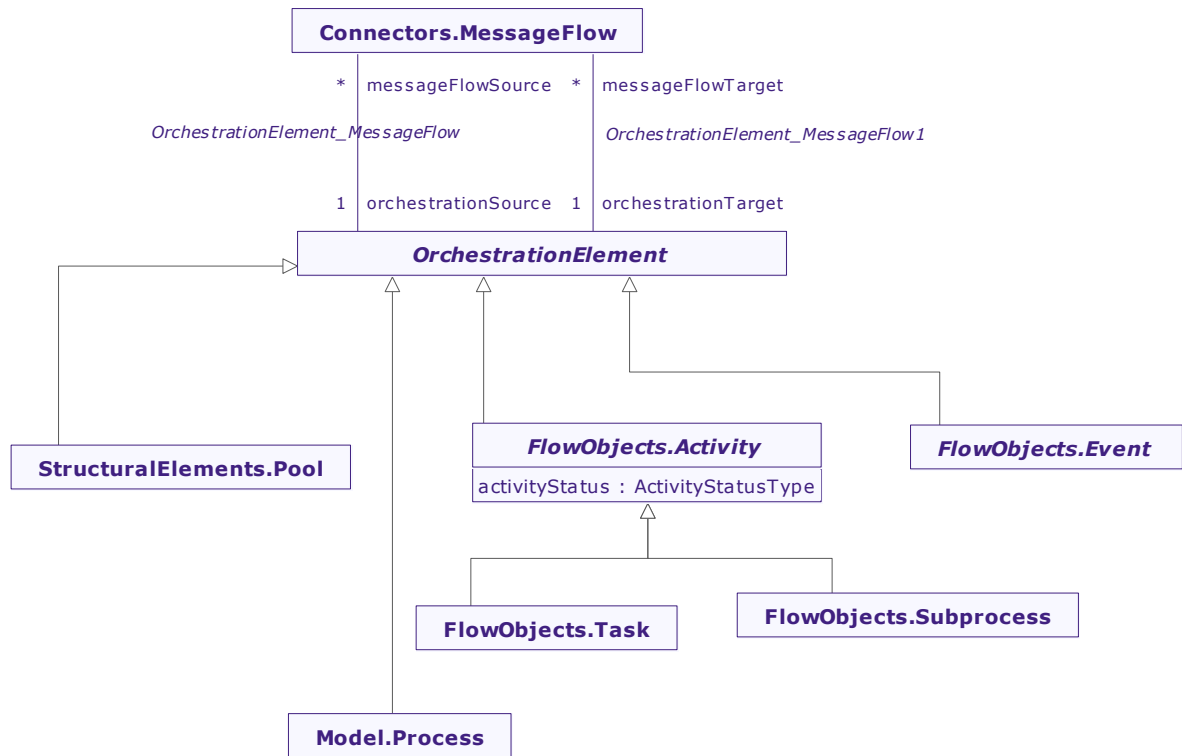
B. Metamodelo BPMN e restrições OCL

a) Metamodelo BPMN

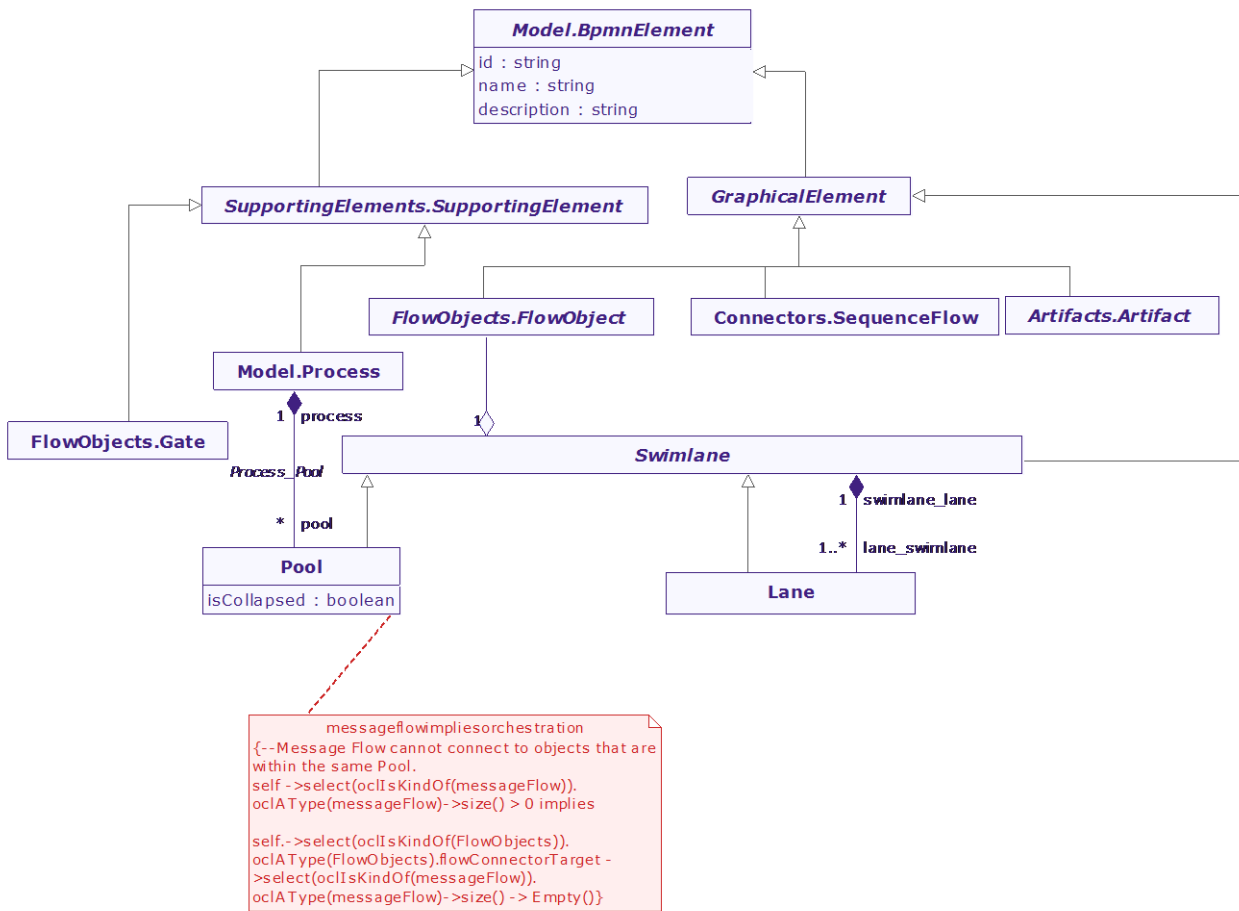
i. Classe Model



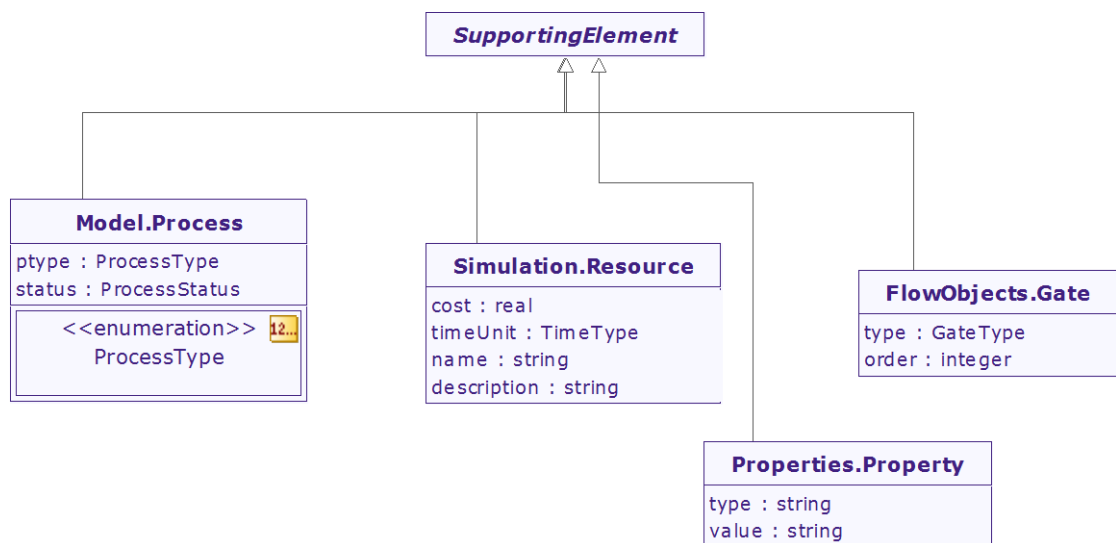
ii. Classe Orchestration



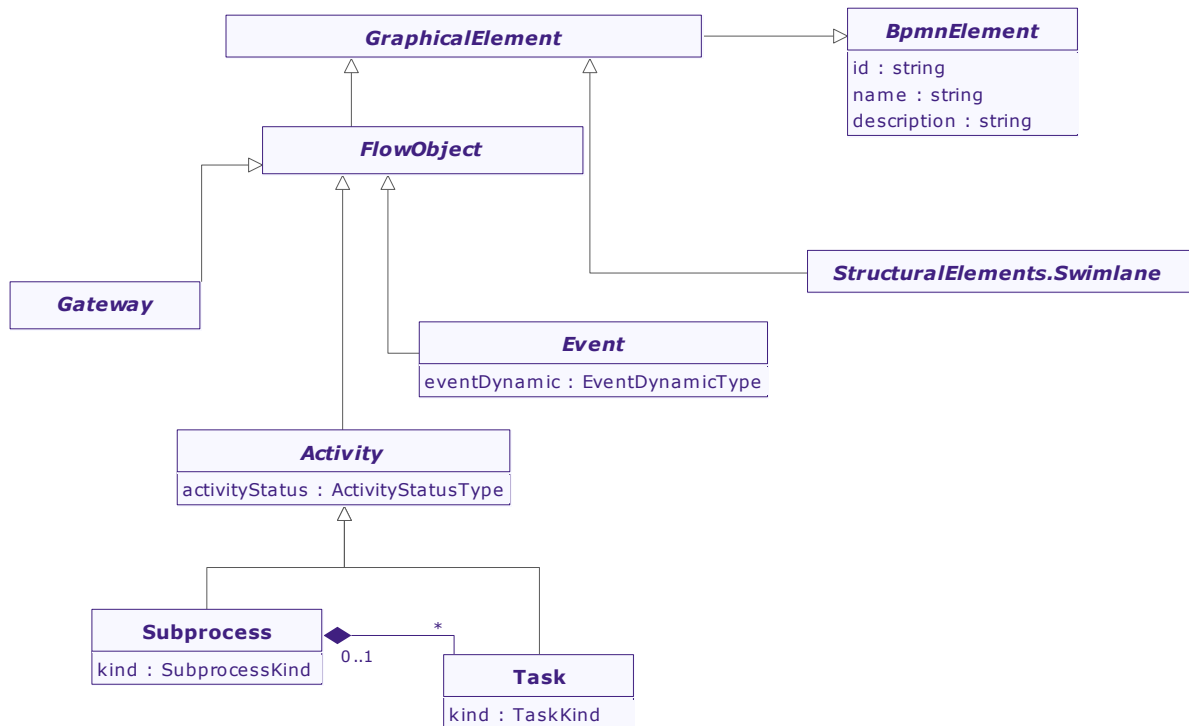
iii. Classe StructuralElements



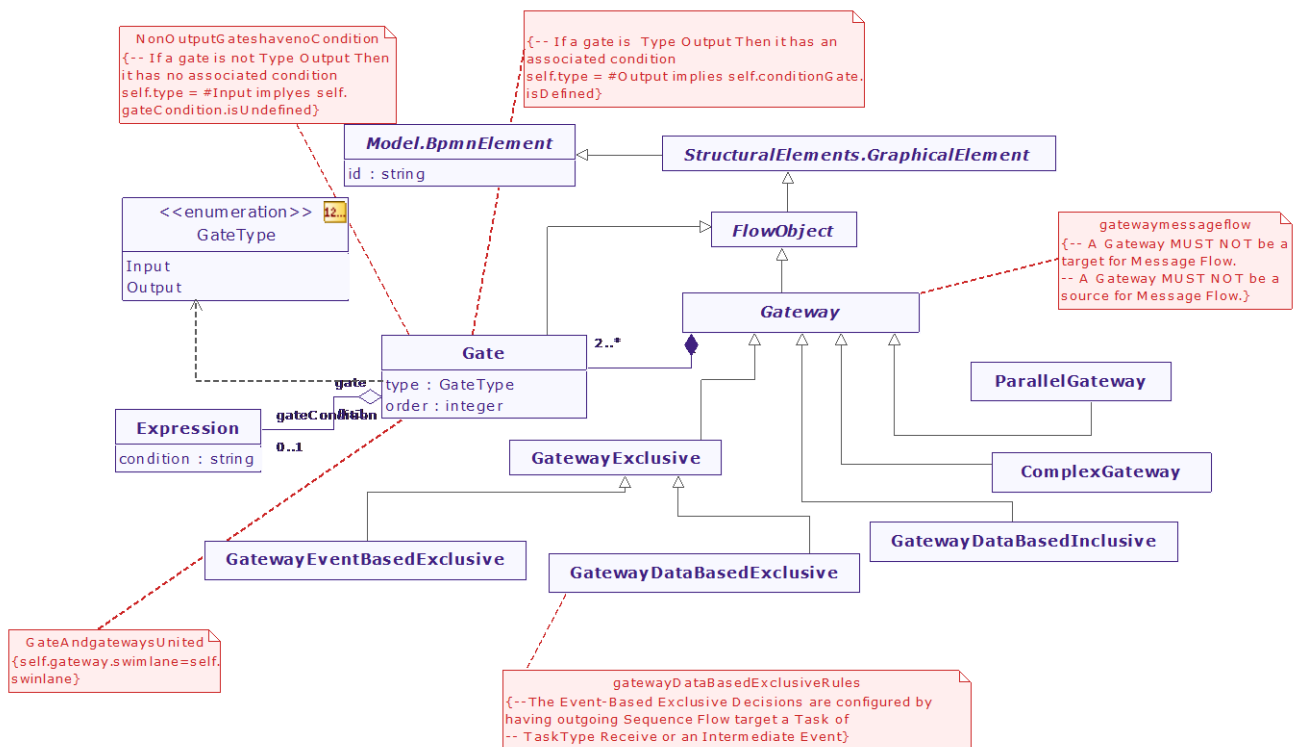
iv. Classe SupportingElements



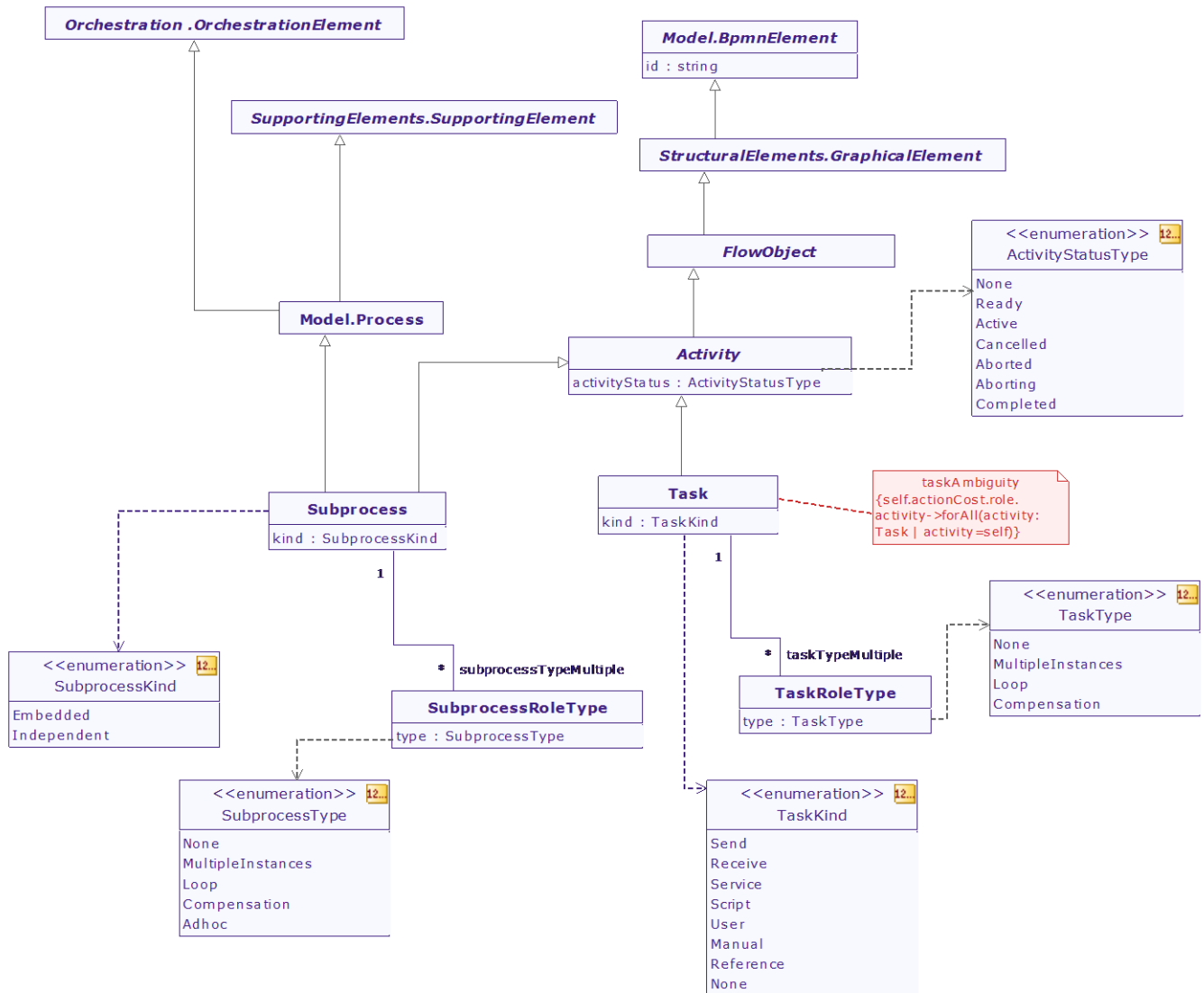
v. Classe FlowObjects



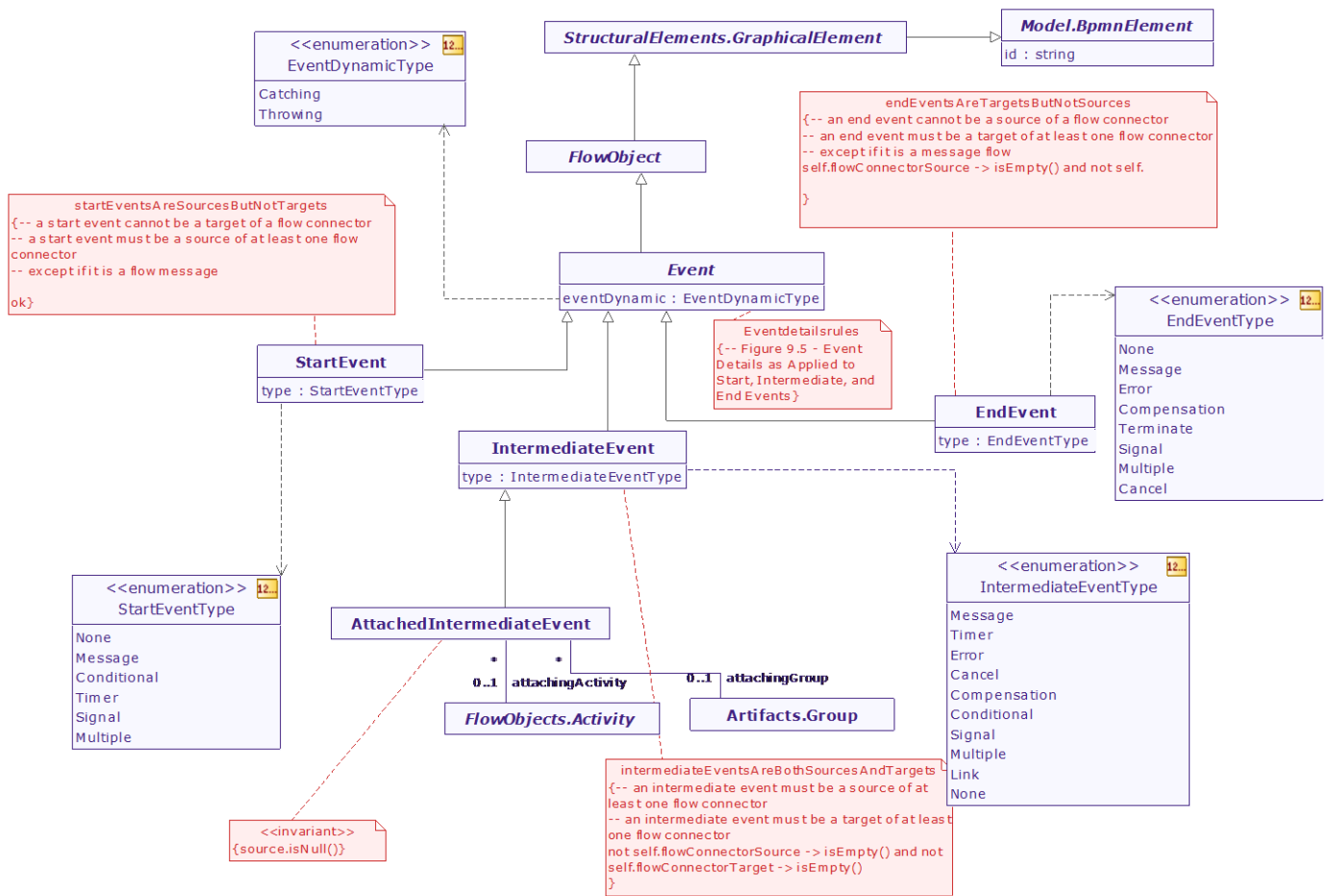
vi. Classe Gateway



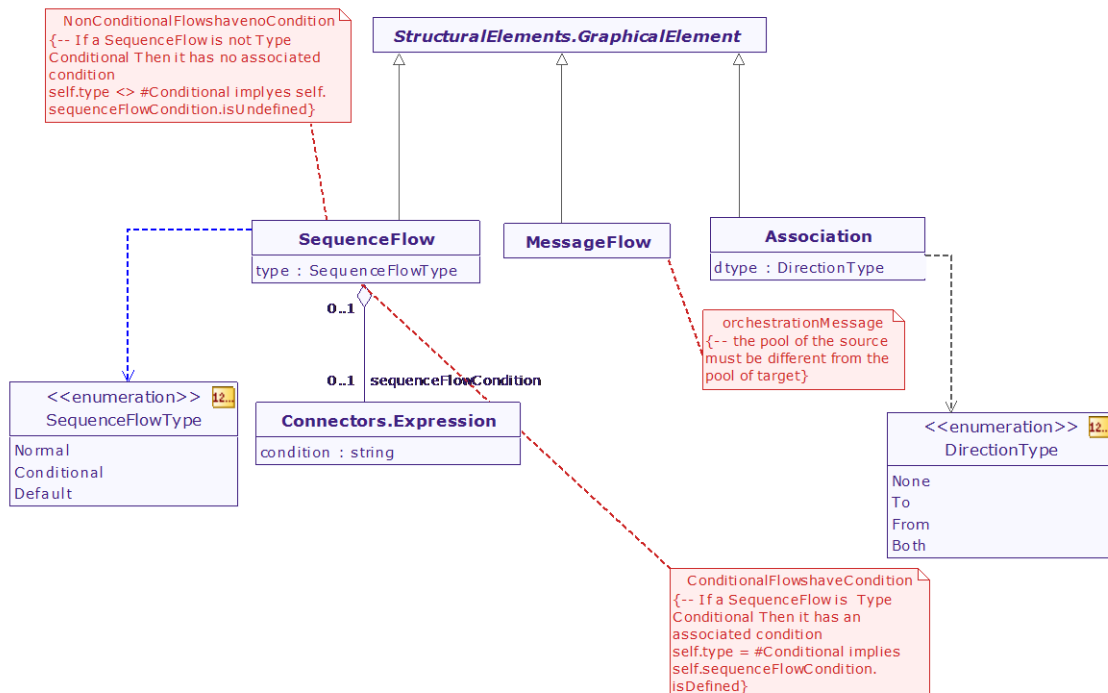
vii. Classe Activities



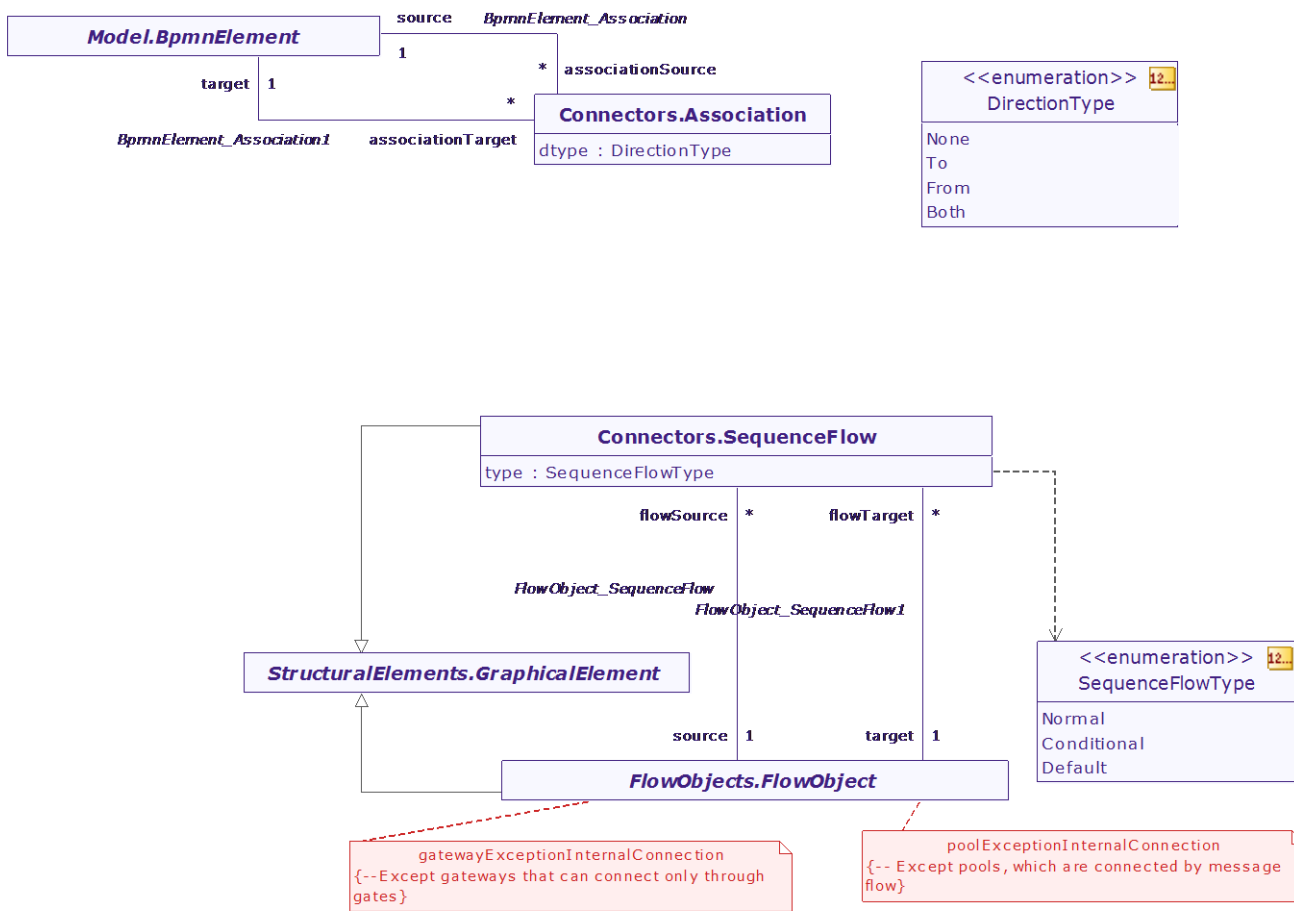
viii. Classe Event



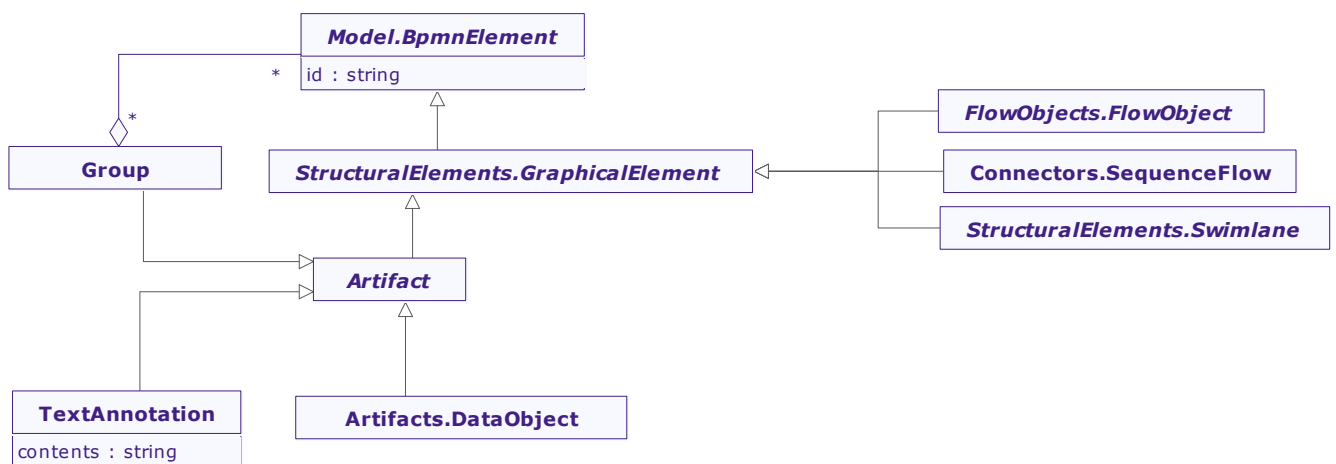
ix. Classe Connectors



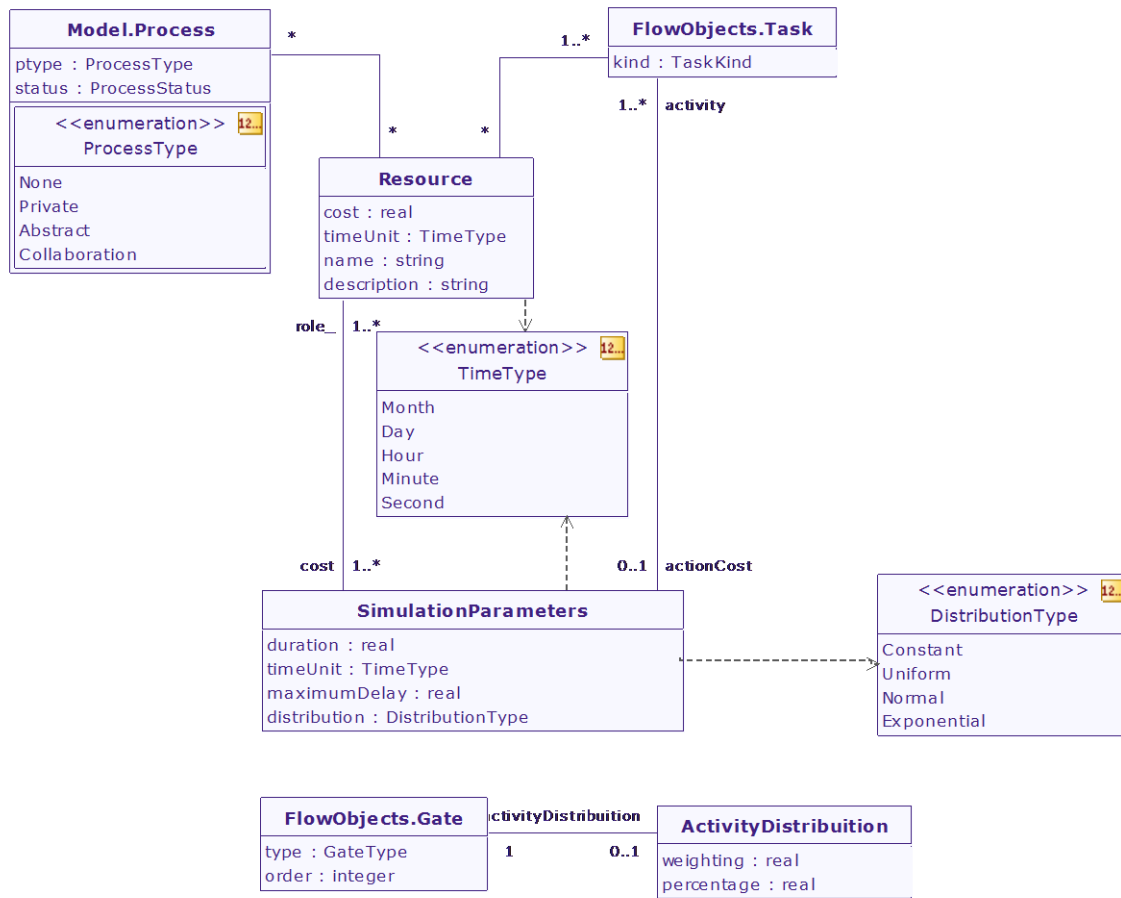
x. Classe Internal Connection



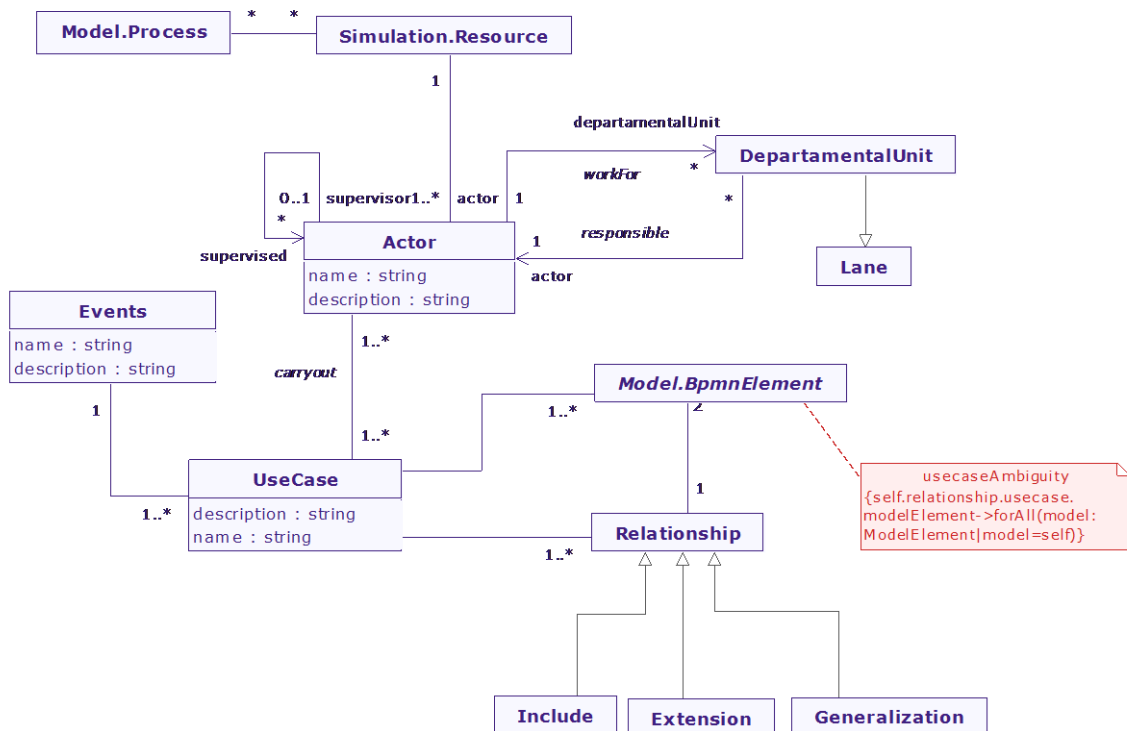
xi. Classe Artifacts



xii. Classe Simulation



xiii. Classe UseCases



b) Métricas OCL

- Métricas para calcular o Tamanho de um Processo

Nome da métrica	Descrição da métrica
countBpmnElements()	Retorna o número de elementos de um processo
totalNumberProcessSwimlanes()	Retorna o número total de swimlanes (<i>pools + lanes</i>) de um processo
totalNumberProcessPools()	Retorna o número total de <i>pools</i> de um processo
totalNumberProcessLanes()	Retorna o número total de <i>lanes</i> de um processo
totalNumberProcessFlowObjects()	Retorna o número total de <i>FlowObjects</i> (gateways, actividades, eventos) de um processo
totalNumberProcessActivities()	Retorna o número total de actividades (subprocessos + <i>tasks</i>) de um processo
totalNumberProcessSubProcesses()	Retorna o número total de subprocessos de um processo
totalNumberProcessTasks()	Retorna o número total de <i>tasks</i> de um processo
totalNumberProcessEvents()	Retorna o número total de eventos (<i>startevent, intermediateevent, endevent</i>) de um processo
totalNumberProcessStartEvents()	Retorna o número total de <i>StartEvents</i> de um processo
totalNumberProcessEndEvents()	Retorna o número total de <i>EndEvents</i> de um processo
totalNumberProcessIntermediateEvents()	Retorna o número total de <i>IntermediateEvents</i> de um processo
totalNumberProcessGateways()	Retorna o número total de <i>gateways</i> (<i>gatewayEventbased, gatewaydatabasedexclusive, gatewaydatabasedinclusive, complexgateway e parallelgateway</i>) de um processo
totalNumberProcessGatewayDataBasedExclusives()	Retorna o número total de <i>GatewayDataBasedExclusives</i> de um processo
totalNumberProcessGatewayEventBasedExclusives()	Retorna o número total de <i>GatewayEventBasedExclusives</i> de um processo
totalNumberProcessGatewayDataBasedInclusives()	Retorna o número total de <i>GatewayDataBasedInclusives</i> de um processo
totalNumberProcessComplexGateways()	Retorna o número total de <i>ComplexGateways</i> de um processo
totalNumberProcessParallelGateways()	Retorna o número total de <i>ParallelGateways</i> de um processo
totalNumberProcessGates()	Retorna o número total de Gates (<i>Input e Output</i>) de um processo
totalNumberProcessInputGates()	Retorna o número total de Gates de <i>Input</i> de um processo
totalNumberProcessOutputGates()	Retorna o número total de Gates de <i>Output</i> de um processo
totalNumberProcessArtifacts()	Retorna o número total de Artefactos (<i>dataObjects, textAnnotations e groups</i>) de um processo
totalNumberProcessDataObjects()	Retorna o número total de <i>DataObjects</i> de um processo
totalNumberProcessTextAnnotations()	Retorna o número total de <i>TexAnnotations</i> de um processo
totalNumberProcessGroups()	Retorna o número total de <i>Groups</i> de um processo
totalNumberProcessConnectors()	Retorna o número total de Conectores (<i>SequenceFlows, MessageFlows e Associations</i>) de um processo
totalNumberProcessSequenceFlows()	Retorna o número total de <i>SequenceFlows</i> de um processo
totalNumberProcessMessageFlows()	Retorna o número total de <i>MessageFlows</i> de um processo
totalNumberProcessAssociations()	Retorna o número total de <i>Associations</i> de um processo

Tabela B.42: Métricas para calcular o tamanho de um processo

Context: class Model_Process

-- Mostra todos os elementos de um Processo

```
bpmnElements(): Set(Model_BpmnElement) = self.pool ->
  iterate(elem: StructuralElements_Pool; acc: Set(Model_BpmnElement) = oclEmpty(Set(Model_BpmnElement)) |
    acc -> union(elem.bpmnElements()))
```

-- Conta o número de elementos de um Processo

```
countBpmnElements(): Integer = self.pool ->
  iterate(elem: StructuralElements_Pool; acc: Set(Model_BpmnElement) = oclEmpty(Set(Model_BpmnElement)) |
    acc -> union(elem.bpmnElements())) -> size()
```

Context: class Model_Process

-- Número total de pools de um processo

```
totalNumberProcessPools(): Integer = self.pool -> select(oclIsKindOf (StructuralElements_Pool)) ->
  collect(oclAsType(StructuralElements_Pool)) -> size()
```

-- Mostra todas as pools de um processo

```
totalProcessPools(): Set(StructuralElements_Pool) = self.pool -> select(oclIsKindOf(StructuralElements_Pool)) ->
  collect(oclAsType(StructuralElements_Pool)) -> asSet()
```

-- Número total de lanes de um processo

```
totalNumberProcessLanes(): Integer = self.pool -> select(oclIsKindOf(StructuralElements_Pool)) ->
  collect(oclAsType(StructuralElements_Pool)).numberLanes() -> size()
```

-- Mostra todas as lanes de um processo

```
totalProcessLanes(): Set(StructuralElements_Lane) = self.pool -> select(oclIsKindOf(StructuralElements_Pool)) ->
  collect(oclAsType(StructuralElements_Pool)).showLanes() -> asSet()
```

-- Número total de swimlanes (pools + lanes) de um processo

```
totalNumberProcessSwimlanes(): Integer = totalNumberProcessPools() + totalNumberProcessLanes()
```


Context: class Model_Process

-- Número total de FlowObjects (gateways, activities e events) de um processo

```
totalNumberProcessFlowObjects(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_FlowObject)) ->size()
```

-- Mostra todos os flowObjects de um processo

```
totalProcessFlowObjects(): Set(FlowObjects_FlowObject) = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_FlowObject))->
    collect(oclAsType(FlowObjects_FlowObject)) -> asSet()
```

-- Número total de actividades (subprocessos + tasks) de um processo

```
totalNumberProcessActivities(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_Activity))->size()
```

-- Mostra todas as actividades de um processo

```
totalProcessActivities(): Set(FlowObjects_Activity) = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_Activity))->
    collect(oclAsType(FlowObjects_Activity)) -> asSet()
```

-- Número total de subprocessos de um processo

```
totalNumberProcessSubProcesses(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_SubProcess))->size()
```

-- Mostra todos os subprocessos de um processo

```
totalProcessSubProcesses(): Set(FlowObjects_SubProcess) = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_SubProcess))->
    collect(oclAsType(FlowObjects_SubProcess)) -> asSet()
```

-- Número total de tasks de um processo

```
totalNumberProcessTasks(): Integer = self.bpmnElements()->select(oclIsKindOf(FlowObjects_Task))->size()
```

-- Mostra todas as tasks de um processo

```
totalProcessTasks(): Set(FlowObjects_Task) = self.bpmnElements()-> select(oclIsKindOf(FlowObjects_Task))->
    collect(oclAsType(FlowObjects_Task)) -> asSet()
```

Contex: class Model_Process

-- Número total de eventos (startevent, intermediateevent, endevent) de um processo

totalNumberProcessEvents(): Integer = self.bpmnElements()->select(oclIsKindOf(FlowObjects_Event))->size()

-- Mostra todos os Eventos de um processo

totalProcessEvents(): Set(FlowObjects_Event) = self.bpmnElements() ->
select(oclIsKindOf(FlowObjects_Event)) ->
collect(oclAsType(FlowObjects_Event)) -> asSet()

-- Número total de StartEvents de um processo

totalNumberProcessStartEvents(): Integer = self.bpmnElements() ->
select(oclIsKindOf(FlowObjects_StartEvent))->size()

-- Mostra todos os StartEventos de um processo

totalProcessStartEvents(): Set(FlowObjects_StartEvent) = self.bpmnElements() ->
select(oclIsKindOf(FlowObjects_StartEvent)) ->
collect(oclAsType(FlowObjects_StartEvent)) -> asSet()

-- Número total de EndEvents de um processo

totalNumberProcessEndEvents(): Integer = self.bpmnElements() ->
select(oclIsKindOf(FlowObjects_EndEvent))->size()

-- Mostra todos os EndEventos de um processo

totalProcessEndEvents(): Set(FlowObjects_EndEvent) = self.bpmnElements() ->
select(oclIsKindOf(FlowObjects_EndEvent))->
collect(oclAsType(FlowObjects_EndEvent)) -> asSet()

-- Número total de IntermediateEvents de um processo

totalNumberProcessIntermediateEvents(): Integer = self.bpmnElements() ->
select(oclIsKindOf(FlowObjects_IntermediateEvent))->size()

-- Mostra todos os IntermediateEvents de um processo

totalProcessIntermediateEvents(): Set(FlowObjects_IntermediateEvent) = self.bpmnElements() ->
select(oclIsKindOf(FlowObjects_IntermediateEvent)) ->
collect(oclAsType(FlowObjects_IntermediateEvent)) -> asSet()

Context: class Model_Process

-- Número total de gateways (GatewayEventbased, gatewaydatabasedexclusive, gatewaydatabasedinclusive, complexgateway e parallelgateway) de um processo

```
totalNumberProcessGateways(): Integer = self.bpmnElements()->
    select(oclIsKindOf(FlowObjects_Gateway))->size()
```

-- Mostra todas as gateways de um processo

```
totalProcessGateways(): Set(FlowObjects_Gateway) = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_Gateway)) ->
    collect(oclAsType(FlowObjects_Gateway)) -> asSet()
```

-- Número total de GatewayDataBasedExclusives de um processo

```
totalNumberProcessGatewayDataBasedExclusives(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_GatewayDataBasedExclusive))->size()
```

-- Mostra todos as GatewayDataBasedExclusives de um processo

```
totalProcessGatewayDataBasedExclusives(): Set(FlowObjects_GatewayDataBasedExclusive) =
    self.bpmnElements()->select(oclIsKindOf(FlowObjects_GatewayDataBasedExclusive)) ->
    collect(oclAsType(FlowObjects_GatewayDataBasedExclusive)) -> asSet()
```

-- Número total de GatewayEventBasedExclusives de um processo

```
totalNumberProcessGatewayEventBasedExclusives(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_GatewayEventBasedExclusive))->size()
```

-- Mostra todos as GatewayEventBasedExclusives de um processo

```
totalProcessGatewayEventBasedExclusives(): Set(FlowObjects_GatewayEventBasedExclusive) =
    self.bpmnElements()->select(oclIsKindOf(FlowObjects_GatewayEventBasedExclusive)) ->
    collect(oclAsType(FlowObjects_GatewayEventBasedExclusive)) -> asSet()
```

-- Número total de GatewayDataBasedInclusives de um processo

```
totalNumberProcessGatewayDataBasedInclusives(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_GatewayDataBasedInclusive))->size()
```

-- Mostra todos as GatewayDataBasedInclusives de um processo

```
totalProcessGatewayDataBasedInclusives(): Set(FlowObjects_GatewayDataBasedInclusive) =
    self.bpmnElements()->select(oclIsKindOf(FlowObjects_GatewayDataBasedInclusive))->
    collect(oclAsType(FlowObjects_GatewayDataBasedInclusive)) -> asSet()
```

-- Número total de ComplexGateways de um processo

```
totalNumberProcessComplexGateways(): Integer = self.bpmnElements()->
    select(oclIsKindOf(FlowObjects_ComplexGateway))->size()
```

-- Mostra todos as ComplexGateways de um processo

```
totalProcessComplexGateways(): Set(FlowObjects_ComplexGateway) = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_ComplexGateway))->
    collect(oclAsType(FlowObjects_ComplexGateway)) -> asSet()
```

-- Número total de ParallelGateways de um processo

```
totalNumberProcessParallelGateways(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_ParallelGateway))->size()
```

-- Mostra todos as ParallelGateways de um processo

```
totalProcessParallelGateways(): Set(FlowObjects_ParallelGateway) = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_ParallelGateway))->
    collect(oclAsType(FlowObjects_ParallelGateway)) -> asSet()
```

Context: class Model_Process

-- Número total de Gates de um processo

```
totalNumberProcessGates(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_Gateway)).oclAsType(FlowObjects_Gateway) ->
    asSet ()-> collect(numberInputGates() + numberOutputGates())->sum()
```

-- Mostra todas as Gates de um processo

```
totalProcessGates(): Set(FlowObjects_FlowObject) = self.bpmnElements()->
    select(oclIsKindOf(FlowObjects_Gateway)) ->
    collect(oclAsType(FlowObjects_Gateway)).Allsuccessors() -> asSet()
```

-- Número total de Gates Input de um processo

```
totalNumberProcessInputGates(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_Gateway)).oclAsType(FlowObjects_Gateway) ->
    asSet ()-> collect(numberInputGates())->sum()
```

-- Mostra todas as Input Gates de um processo

```
totalProcessInputGates(): Set(FlowObjects_FlowObject) = self.bpmnElements()->
    select(oclIsKindOf(FlowObjects_Gateway)) ->
    collect(oclAsType(FlowObjects_Gateway)).Inputsuccessors() -> asSet()
```

-- Número total de Gates Output de um processo

```
totalNumberProcessOutputGates(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_Gateway)).oclAsType(FlowObjects_Gateway)->
    asSet() ->collect(numberOutputGates())->sum()
```

-- Mostra todas as Output Gates de um processo

```
totalProcessOutputGates(): Set(FlowObjects_FlowObject) = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_Gateway))->
    collect(oclAsType(FlowObjects_Gateway)).successors() -> asSet()
```

Context: class Model_Process

-- Número total de Artifacts (DataObjects, TextAnnotations and Groups) de um processo

totalNumberProcessArtifacts(): Integer = self.bpmnElements()->select(oclIsKindOf(Artifacts_Artifact))->size()

-- Mostra todos os Artifacts de um processo

totalProcessArtifacts(): Set(Artifacts_Artifact) = self.bpmnElements()-> select(oclIsKindOf(Artifacts_Artifact))
-> collect(oclAsType(Artifacts_Artifact)) -> asSet()

-- Número total de DataObjects de um processo

totalNumberProcessDataObjects(): Integer = self.bpmnElements() ->
select(oclIsKindOf(Artifacts_DataObject)) -> size()

-- Mostra todos os DataObjects de um processo

totalProcessDataObjects(): Set(Artifacts_DataObject) = self.bpmnElements() ->
select(oclIsKindOf(Artifacts_DataObject)) ->
collect(oclAsType(Artifacts_DataObject)) -> asSet()

-- Número total de TexAnnotations de um processo

totalNumberProcessTextAnnotations(): Integer = self.bpmnElements() ->
select(oclIsKindOf(Artifacts_TextAnnotation))->size()

-- Mostra todos os TexAnnotations de um processo

totalProcessTextAnnotations(): Set(Artifacts_TextAnnotation) = self.bpmnElements() ->
select(oclIsKindOf(Artifacts_TextAnnotation)) ->
collect(oclAsType(Artifacts_TextAnnotation)) -> asSet()

-- Número total de Groups de um processo

totalNumberProcessGroups(): Integer = self.bpmnElements()->select(oclIsKindOf(Artifacts_Group)) -> size()

-- Mostra todos os Groups de um processo

totalProcessGroups(): Set(Artifacts_Group) = self.bpmnElements()->select(oclIsKindOf(Artifacts_Group)) ->
collect(oclAsType(Artifacts_Group)) -> asSet()

Context: class Model_Process

-- Número total de SequenceFlows de um processo

```
totalNumberProcessSequenceFlows(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(Connectors_SequenceFlow)) -> size()
```

-- Mostra todos os SequenceFlows de um processo

```
totalProcessSequenceFlows(): Set(Connectors_SequenceFlow) = self.bpmnElements() ->
    select(oclIsKindOf(Connectors_SequenceFlow)) ->
    collect(oclAsType(Connectors_SequenceFlow)) -> asSet()
```

-- Número total de MessageFlows de um processo

```
totalNumberProcessMessageFlows(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(Connectors_MessageFlow)) -> size()
```

-- Mostra todos os MessageFlows de um processo

```
totalProcessMessageFlows(): Set(Connectors_MessageFlow) = self.bpmnElements()->
    select(oclIsKindOf(Connectors_MessageFlow)) ->
    collect(oclAsType(Connectors_MessageFlow)) -> asSet()
```

-- Número total de Associations de um processo

```
totalNumberProcessAssociations(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(Connectors_Association)) ->size()
```

-- Mostra todos os Associations de um processo

```
totalProcessAssociations(): Set(Connectors_Association) = self.bpmnElements() ->
    select(oclIsKindOf(Connectors_Association))->
    collect(oclAsType(Connectors_Association)) -> asSet()
```

-- Número total de Connectores (SequenceFlows, MessageFlows e Associations) de um processo

```
totalNumberProcessConnectors(): Integer = self.totalNumberProcessSequenceFlows() +
    self.totalNumberProcessMessageFlows() + self.totalNumberProcessAssociations()
```

- **Métricas para calcular o Tamanho de um Modelo**

Nome da métrica	Descrição da métrica
totalNumberModelProcesses()	Retorna o número total de processos de um modelo
totalNumberModelSwimlanes()	Retorna o número total de swimlanes (<i>pools + lanes</i>) de um modelo
totalNumberModelPools()	Retorna o número total de <i>pools</i> de um modelo
totalNumberModelLanes()	Retorna o número total de <i>lanes</i> de um modelo
totalNumberModelFlowObjects()	Retorna o número total de <i>FlowObjects</i> (gateways, actividades, eventos) de um modelo
totalNumberModelActivities()	Retorna o número total de actividades (subprocessos + <i>tasks</i>) de um modelo
totalNumberModelSubProcesses()	Retorna o número total de subprocessos de um modelo
totalNumberModelTasks()	Retorna o número total de <i>tasks</i> de um modelo
totalNumberModelEvents()	Retorna o número total de eventos (<i>startevent, intermediateevent, endevent</i>) de um modelo
totalNumberModelStartEvents()	Retorna o número total de <i>StartEvents</i> de um modelo
totalNumberModelEndEvents()	Retorna o número total de <i>EndEvents</i> de um modelo
totalNumberModelIntermediateEvents()	Retorna o número total de <i>IntermediateEvents</i> de um modelo
totalNumberModelGateways()	Retorna o número total de <i>gateways</i> (<i>gatewayEventbased, gatewaydatabasedexclusive, gatewaydatabasedinclusive, complexgateway e parallelgateway</i>) de um modelo
totalNumberModelGatewayDataBasedExclusives()	Retorna o número total de <i>GatewayDataBasedExclusives</i> de um modelo
totalNumberModelGatewayEventBasedExclusives()	Retorna o número total de <i>GatewayEventBasedExclusives</i> de um modelo
totalNumberModelGatewayDataBasedInclusives()	Retorna o número total de <i>GatewayDataBasedInclusives</i> de um modelo
totalNumberModelComplexGateways()	Retorna o número total de <i>ComplexGateways</i> de um modelo
totalNumberModelParallelGateways()	Retorna o número total de <i>ParallelGateways</i> de um modelo
totalNumberModelGates()	Retorna o número total de Gates (<i>Input e Output</i>) de um modelo
totalNumberModelInputGates()	Retorna o número total de Gates de <i>Input</i> de um modelo
totalNumberModelOutputGates()	Retorna o número total de Gates de <i>Output</i> de um modelo
totalNumberModelArtifacts()	Retorna o número total de Artefactos (<i>dataObjects, textAnnotations e groups</i>) de um modelo
totalNumberModelDataObjects()	Retorna o número total de <i>DataObjects</i> de um modelo
totalNumberModelTextAnnotations()	Retorna o número total de <i>TexAnnotations</i> de um modelo
totalNumberModelGroups()	Retorna o número total de <i>Groups</i> de um modelo
totalNumberModelConnectors()	Retorna o número total de Conectores (<i>SequenceFlows, MessageFlows e Associations</i>) de um modelo
totalNumberModelSequenceFlows()	Retorna o número total de <i>SequenceFlows</i> de um modelo
totalNumberModelMessageFlows()	Retorna o número total de <i>MessageFlows</i> de um modelo
totalNumberModelAssociations()	Retorna o número total de <i>Associations</i> de um modelo

Tabela B.43: Métricas para calcular o tamanho de um modelo

Context: class Model_Model

-- Número total de processos de um modelo

totalNumberModelProcesses(): Integer = self.process -> size()

Context: class Model_Model

-- Número total de swimlanes (pools + lanes) de um modelo

totalNumberModelSwimlanes(): Integer = self.process -> collect(totalNumberProcessSwimlanes())->sum()

-- Número total de pools de um modelo

totalNumberModelPools(): Integer = self.process -> collect(totalNumberProcessPools()) -> sum()

-- Mostra todas as pools de um modelo

totalModelPools(): Set(Set(StructuralElements_Pool)) = self.process -> collect(totalProcessPools()) -> asSet()

-- Número total de lanes de um modelo

totalNumberModelLanes(): Integer = self.process -> collect(totalNumberProcessLanes())->sum()

-- Mostra todas as lanes de um modelo

totalModelLanes(): Set(Set(StructuralElements_Lane)) = self.process -> collect(totalProcessLanes()) -> asSet()

Context: class Model_Model

-- Número total de FlowObjects (gateways, activities e events) de um modelo

totalNumberModelFlowObjects(): Integer = self.process->collect(totalNumberProcessFlowObjects())->sum()

-- Mostra todos os FlowObjects (gateways, activities e events) de um modelo

totalModelFlowObjects(): Set(Set(FlowObjects_FlowObject))=self.process->collect(totalProcessFlowObjects()) -> asSet()

-- Número total de atividades (subprocessos + tasks) de um modelo

totalNumberModelActivities(): Integer = self.process->collect(totalNumberProcessActivities())->sum()

-- Mostra todas as atividades (subprocessos + tasks) de um modelo

totalModelActivities(): Set(Set(FlowObjects_Activity)) = self.process->collect(totalProcessActivities()) -> asSet()

-- Número total de subprocessos de um modelo

totalNumberModelSubProcesses(): Integer = self.process->collect(totalNumberProcessSubProcesses())->sum()

-- Mostra todos os subprocessos de um modelo

totalModelSubProcesses(): Set(Set(FlowObjects_SubProcess)) = self.process ->
collect(totalProcessSubProcesses()) -> asSet()

-- Número total de tasks de um modelo

totalNumberModelTasks(): Integer = self.process->collect(totalNumberProcessTasks())->sum()

-- Mostra todas as tasks de um modelo

totalModelTasks(): Set(Set(FlowObjects_Task)) = self.process->collect(totalProcessTasks()) -> asSet()

Context: class Model_Model

-- Número total de eventos (startEvent, intermediateEvent, endEvent) de um modelo

totalNumberModelEvents(): Integer = self.process->collect(totalNumberProcessEvents())->sum()

-- Mostra todos os eventos (startEvent, intermediateEvent, endEvent) de um modelo

totalModelEvents(): Set(Set(FlowObjects_Event)) = self.process->collect(totalProcessEvents()) -> asSet()

-- Número total de StartEvents de um modelo

totalNumberModelStartEvents(): Integer = self.process->collect(totalNumberProcessStartEvents())->sum()

-- Mostra todos os StartEvents de um modelo

totalModelStartEvents(): Set(Set(FlowObjects_StartEvent)) = self.process -> collect(totalProcessStartEvents())
-> asSet()

-- Número total de EndEvents de um modelo

totalNumberModelEndEvents(): Integer = self.process->collect(totalNumberProcessEndEvents())->sum()

-- Mostra todos os EndEvents de um modelo

totalModelEndEvents(): Set(Set(FlowObjects_EndEvent)) = self.process ->
collect(totalProcessEndEvents()) -> asSet()

-- Número total de IntermediateEvents de um modelo

totalNumberModelIntermediateEvents(): Integer = self.process ->
collect(totalNumberProcessIntermediateEvents())->sum()

-- Mostra todos os IntermediateEvents de um modelo

totalModelIntermediateEvents(): Set(Set(FlowObjects_IntermediateEvent)) = self.process ->
collect(totalProcessIntermediateEvents()) -> asSet()

Context: class Model_Model

-- Número total de gateways (GatewayEventbased, gatewaydatabasedexclusive, gatewaydatabasedinclusive, complexgateway e parallelgateway) de um modelo

totalNumberModelGateways(): Integer = self.process->collect(totalNumberProcessGateways())->sum()

-- Mostra todas as Gateways (GatewayEventbased, gatewaydatabasedexclusive, gatewaydatabasedinclusive, complexgateway e parallelgateway) de um modelo

totalModelGateways(): Set(Set(FlowObjects_Gateway)) = self.process->collect(totalProcessGateways()) -> asSet()

-- Número total de GatewayDataBasedExclusives de um modelo

totalNumberModelGatewayDataBasedExclusives(): Integer = self.process ->
collect(totalNumberProcessGatewayDataBasedExclusives())->sum()

-- Mostra todas as GatewayDataBasedExclusives de um modelo

totalModelGatewayDataBasedExclusives(): Set(Set(FlowObjects_GatewayDataBasedExclusive)) =
self.process -> collect(totalProcessGatewayDataBasedExclusives()) -> asSet()

-- Número total de GatewayEventBasedExclusives de um modelo

totalNumberModelGatewayEventBasedExclusives(): Integer = self.process ->
collect(totalNumberProcessGatewayEventBasedExclusives()) -> sum()

-- Mostra todas as GatewayEventBasedExclusives de um modelo

totalModelGatewayEventBasedExclusives(): Set(Set(FlowObjects_GatewayEventBasedExclusive)) =
self.process->collect(totalProcessGatewayEventBasedExclusives()) -> asSet()

-- Número total de GatewayDataBasedInclusives de um modelo

totalNumberModelGatewayDataBasedInclusives(): Integer = self.process ->
collect(totalNumberProcessGatewayDataBasedInclusives())->sum()

-- Mostra todas as GatewayDataBasedInclusives de um modelo

totalModelGatewayDataBasedInclusives(): Set(Set(FlowObjects_GatewayDataBasedInclusive)) = self.process
-> collect(totalProcessGatewayDataBasedInclusives()) -> asSet()

-- Número total de ComplexGateways de um modelo

totalNumberModelComplexGateways(): Integer = self.process ->
collect(totalNumberProcessComplexGateways()) ->sum()

-- Mostra todas as ComplexGateways de um modelo

totalModelComplexGateways(): Set(Set(FlowObjects_ComplexGateway)) = self.process ->
collect(totalProcessComplexGateways()) -> asSet()

-- Número total de ParallelGateways de um modelo

totalNumberModelParallelGateways(): Integer = self.process ->
collect(totalNumberProcessParallelGateways()) -> sum()

-- Mostra todas as ParallelGateways de um modelo

totalModelParallelGateways(): Set(Set(FlowObjects_ParallelGateway)) = self.process ->
collect(totalProcessParallelGateways()) -> asSet()

Context: class Model_Model

-- Número total de Gates de um modelo

totalNumberModelGates(): Integer = self.process->collect(totalNumberProcessGates())->sum()

-- Mostra todas as Gates(Input e Output) de um modelo

totalModelGates(): Set(Set(FlowObjects_FlowObject)) = self.process->collect(totalProcessGates()) -> asSet()

-- Número total de InputGates de um modelo

totalNumberModelInputGates(): Integer = self.process->collect(totalNumberProcessInputGates())->sum()

-- Mostra todas as InputGates de um modelo

totalModelInputGates(): Set(Set(FlowObjects_FlowObject)) = self.process ->
collect(totalProcessInputGates()) -> asSet()

-- Número total de OutputGates de um modelo

totalNumberModelOutputGates(): Integer = self.process->collect(totalNumberProcessOutputGates())->sum()

-- Mostra todas as InputGates de um modelo

totalModelOutputGates(): Set(Set(FlowObjects_FlowObject)) = self.process ->
collect(totalProcessOutputGates()) -> asSet()

Context: class Model_Model

-- Número total de Artifacts (dataObjects, textAnnotations and groups) de um modelo

totalNumberModelArtifacts(): Integer = self.process->collect(totalNumberProcessArtifacts())->sum()

-- Mostra todos os Artifacts (dataObjects, textAnnotations and groups) de um modelo

totalModelArtifacts(): Set(Set(Artifacts_Artifact)) = self.process->collect(totalProcessArtifacts()) -> asSet()

-- Número total de DataObjects de um modelo

totalNumberModelDataObjects(): Integer = self.process->collect(totalNumberProcessDataObjects())->sum()

-- Mostra todos os DataObjects de um modelo

totalModelDataObjects(): Set(Set(Artifacts_DataObject)) = self.process->collect(totalProcessDataObjects()) -> asSet()

-- Número total de TextAnnotations de um modelo

totalNumberModelTextAnnotations(): Integer = self.process->collect(totalNumberProcessTextAnnotations())->sum()

-- Mostra todos os TextAnnotations de um modelo

totalModelTextAnnotations(): Set(Set(Artifacts_TextAnnotation)) = self.process ->
collect(totalProcessTextAnnotations()) -> asSet()

-- Número total de Groups de um modelo

totalNumberModelGroups(): Integer = self.process->collect(totalNumberProcessGroups())->sum()

-- Mostra todos os Groups de um modelo

totalModelGroups(): Set(Set(Artifacts_Group)) = self.process->collect(totalProcessGroups()) -> asSet()

Context: class Model_Model

-- Número total de Connectores de um modelo

totalNumberModelConnectors(): Integer = self.process -> collect(totalNumberProcessConnectors())->sum()

-- Número total de SequenceFlows de um modelo

totalNumberModelSequenceFlows(): Integer = self.process->collect(totalNumberProcessSequenceFlows())-> sum()

-- Mostra todos os SequenceFlows de um modelo

totalModelSequenceFlows(): Set(Set(Connectors_SequenceFlow)) = self.process ->
collect(totalProcessSequenceFlows()) -> asSet()

-- Número total de MessageFlows de um modelo

totalNumberModelMessageFlows(): Integer = self.process -> collect(totalNumberProcessMessageFlows()) -> sum()

-- Mostra todos os MessageFlows de um modelo

totalModelMessageFlows(): Set(Set(Connectors_MessageFlow)) = self.process ->
collect(totalProcessMessageFlows()) -> asSet()

-- Número total de Associations de um modelo

totalNumberModelAssociations(): Integer = self.process->collect(totalNumberProcessAssociations())->sum()

-- Mostra todas as Associations de um modelo

totalModelAssociations(): Set(Set(Connectors_Association)) = self.process ->
collect(totalProcessAssociations()) -> asSet()

- **Métricas para calcular a Complexidade de um Modelo**

Métrica CFC de um processo

Context: class Model_Process

-- Calcula os XORs Data-based de um processo

```
cFC_XOR_DataBased(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_GatewayDataBasedExclusive)).
    oclAsType(FlowObjects_GatewayDataBasedExclusive)->asSet() ->
    select(isSplit()) -> collect(numberOutputGates()) -> sum()
```

-- Calcula os XORs Event-based de um processo

```
cFC_XOR_EventBased(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_GatewayEventBasedExclusive)).
    oclAsType(FlowObjects_GatewayEventBasedExclusive)->asSet()->
    select(isSplit()) -> collect(numberOutputGates()) -> sum()
```

-- Calcula os ORs (DataBasedInclusivede) um processo

```
cFC_OR(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_GatewayDataBasedInclusive)).
    oclAsType(FlowObjects_GatewayDataBasedInclusive)->asset() ->
    select(isSplit()) -> collect(numberOutputGates()) -> sum()
```

-- Calcula os ANDs (ParallelGateway) de um processo

```
cFC_AND(): Integer = self.bpmnElements() ->
    select(oclIsKindOf(FlowObjects_ParallelGateway)).
    oclAsType(FlowObjects_ParallelGateway) ->asset() ->
    select(isSplit()) -> collect(numberOutputGates()) -> sum()
```

-- Métrica CFC - calcula a complexidade de um processo adicionando os XORs, ORs e ANDs dos processos

```
complexityProcessCFC (): Integer = cFC_XOR_DataBased() + cFC_XOR_EventBased() +
    cFC_OR() + cFC_AND()
```

Métrica CFC de um modelo

Context: class Model_Model

-- Métrica CFC - calcula a complexidade de um modelo, adicionando os XORs, ORs e ANDs dos processos

```
complexityModelCFC (): Integer = self.process -> collect(cFC_XOR_DataBased() +
    cFC_XOR_EventBased() + cFC_OR() + cFC_AND()) -> sum()
```

Métrica HPC de um processo

Context: class Model_Process

-- **n1 = número de atividades, splits e joins únicos**

existActivities(): Integer = if totalNumberProcessActivities() >= 1 then 1 else 0 endif

existSplits(): Integer = if totalNumberProcessOutputGates() > 1 then 1 else 0 endif

existJoins(): Integer = if totalNumberProcessInputGates() > 1 then 1 else 0 endif

n1(): Integer = existActivities() + existSplits() + existJoins()

-- **n2 = número de variáveis de dados únicas (dataObject)**

existDataObjects(): Integer = if totalNumberProcessDataObjects() >= 1 then 1 else 0 endif

existTextAnnotations(): Integer = if totalNumberProcessTextAnnotations() > 1 then 1 else 0 endif

n2(): Integer = existDataObjects() + existTextAnnotations()

-- **N1 = número total de atividades, splits e joins de um processo**

N1(): Integer = totalNumberProcessActivities() + totalNumberProcessGateways()

-- **N2 = número total de variáveis de dados de um processo**

N2(): Integer = totalNumberProcessDataObjects() + totalNumberProcessTextAnnotations()

-- **Comprimento do processo: $N=n1*\log_2(n1) + n2*\log_2(n2)$**

length(): Real = n1() * logarithm(2, n1()) + n2() * logarithm(2, n2())

-- **Volume do processo: $V=(N1 + N2)*\log_2(n1+n2)$**

volume(): Real = (N1() + N2()) * logarithm(2, n1()+n2())

-- **Dificuldade do processo: $D=(n1/2)*(N2/n2)$**

difficulty(): Real = if (n2() = 0) then n1()/2
 else (n1()/2)*(N2()/n2())
 endif

Métrica HPC de um modelo

Context: class Model_Model

-- **Comprimento do modelo: Para cada processo: $N=n1*\log_2(n1) + n2*\log_2(n2)$**

lengthModel(): Real = self.process -> collect(length()) -> sum()

-- **Volume do modelo: Para cada processo: $V=(N1 + N2)*\log_2(n1+n2)$**

volumeModel(): Real = self.process -> collect(volume()) -> sum()

-- **Dificuldade do modelo: Para cada processo: $D=(n1/2)*(N2/n2)$**

difficultyModel(): Real = self.process -> collect(difficulty()) -> sum()

Métrica CNC de um processo

Context: class Model_Process

-- Métrica CNC (Coefficient of Network Complexity) = total de Connectors / total de FlowObjects (atividades, eventos e gateways) de um processo

complexityProcessCNC () : Real = totalNumberProcessSequenceFlows() / totalNumberProcessFlowObjects()

Métrica CNC de um modelo

Context: class Model_Model

-- Métrica CNC (Coefficient of Network Complexity) = total de Connectors / total de FlowObjects (atividades, eventos e gateways) de um modelo

complexityModelCNC (): Real = self.process -> collect(complexityProcessCNC()) -> sum()

-- ou

complexityModelCNC (): Real = totalNumberModelSequenceFlows() / totalNumberModelFlowObjects()

- Métricas para calcular a Estrutura de um Modelo

Métrica *nesting depth* de um processo

Context: class Model_Process

-- A Métrica Nesting depth deve percorrer um dado caminho com um início e um destino e contar o número de gateways

nestingDepth(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
 compute¹(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gateway)) ->
 collect(oclAsType(FlowObjects_Gateway)) -> size

-- A Métrica Complexidade BPM auxiliar que percorre um dado caminho com uma origem e um destino e conta o número de gates de output

cfcAux(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
 compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate)) ->
 collect(oclAsType(FlowObjects_Gate)) -> select(type=#Output) -> size

--Métrica Complexidade BPM = Nesting depth + CFC

complexityBPM(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
 nestingDepth(origin, destination) + cfcAux(origin, destination)

¹ – A função **compute (origin, destination)** está definida na subsecção “5.5.1.5 Métricas Nossas”

- **Métricas para calcular a Modularização de um Modelo**

Métrica HKM de um processo

Contex: class Model_Process

-- **Métrica HKM = número total de actividades (tasks e subprocessos) x (nº de startevents (fan-in) x nº de endevents (fan-out))^2**

modularizationProcessHKM(): Real = totalNumberProcessActivities() *
power((totalNumberProcessStartEvents() * totalNumberProcessEndEvents()), 2)

Métrica HKM de um modelo

Contex: class Model_Model

-- **Métrica HKM = número total de actividades (tasks e subprocessos) x (nº de startevents (fan-in) x nº de endevents (fan-out))^2**

modularizationModelHKM(): Real = self.process -> collect(modularizationProcessHKM())-> sum()

- **Métricas novas**

Context: class Model_Process

-- Função que calcula todos os caminhos possíveis, dando a origem e o destino

compute(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :

```
Set(Sequence(FlowObjects_FlowObject)) =
  visit( origin,
        destination,
        oclEmpty(Set(FlowObjects_FlowObject))->including(origin),
        oclEmpty(Sequence(FlowObjects_FlowObject))->append(origin),
        oclEmpty(Set(Sequence(FlowObjects_FlowObject))))
```

```
visit ( x: FlowObjects_FlowObject,
        destination: FlowObjects_FlowObject,
        visited: Set(FlowObjects_FlowObject),
        path: Sequence(FlowObjects_FlowObject),
        paths: Set(Sequence(FlowObjects_FlowObject))) : Set(Sequence(FlowObjects_FlowObject)) =
  if x=destination then paths->including(path)
  else
    if x.successors()->isEmpty() then paths
    else
      x.successors()->iterate(elem: FlowObjects_FlowObject;
                             acc: Set(Sequence(FlowObjects_FlowObject)) = paths |
                             if visited->excludes(elem) then
                               visit(elem, destination, visited->including(elem), path->append(elem), acc)
                             else
                               acc
                             endif)
    endif
  endif
```

Context: class Model_Process

-- Função que conta o número de caminhos possíveis de um processo

numberPaths(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject): Integer =
 compute(origin, destination) -> size()

-- Função que mostra todos os FlowObjects dos vários caminhos possíveis de um processo (análise de cobertura)

allElements(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):

```
Set(FlowObjects_FlowObject) =
  compute(origin, destination) -> flatten
```

-- Função que conta todos os FlowObjects dos vários caminhos possíveis de um processo

countAllElements(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject): Integer =
 compute(origin, destination) -> flatten -> size()

-- Função que conta o número de FlowObjects em todos os caminhos possíveis de um processo

countObjectsPaths(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):

```
Bag(Integer) = compute(origin, destination) ->
  collect(oclIsKindOf(FlowObjects_FlowObject)).count(true)
```

Context: class Model_Process

-- Função que retorna o maior caminho. Em caso de empate retorna o primeiro dos maiores caminhos

biggerPath (origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):

```
Sequence(FlowObjects_FlowObject) =
  compute (origin, destination) -> iterate(elem: Sequence(FlowObjects_FlowObject);
    acc: Sequence(FlowObjects_FlowObject) = oclEmpty(Sequence(FlowObjects_FlowObject))
    if (elem->size() > acc->size()) then
      elem
    else
      acc
    endif)
```

-- Função que conta o número de objectos do maior caminho de um processo

countObjectsBiggerPath(origin: FlowObjects_FlowObject,
destination: FlowObjects_FlowObject): Integer =
 biggerPath(origin, destination) -> size()

-- Função que retorna o menor caminho. Em caso de empate retorna o último dos menores caminhos

smallerPath (origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):

```
Sequence(FlowObjects_FlowObject) =
  compute (origin, destination) -> iterate(elem: Sequence(FlowObjects_FlowObject);
    acc: Sequence(FlowObjects_FlowObject) = biggerPath(origin, destination)|
    if (elem->size() > acc->size()) then
      acc
    else
      elem
    endif)
```

-- Função que conta o número de objectos do menor caminho de um processo

countObjectsSmallerPath(origin: FlowObjects_FlowObject,
destination: FlowObjects_FlowObject): Integer =
 smallerPath(origin, destination) -> size()

Context: class Model_Process

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de actividades

```
countActivities(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Activity))->
    collect(oclAsType(FlowObjects_Activity))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as actividades contidas nesse caminho

```
seeActivities(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
    Set(FlowObjects_Activity) =
        compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Activity))->
        collect(oclAsType(FlowObjects_Activity)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de tasks

```
countTasks(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Task))->
    collect(oclAsType(FlowObjects_Task))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as tasks contidas nesse caminho

```
seeTasks(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Set(FlowObjects_Task) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Task))->
    collect(oclAsType(FlowObjects_Task)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de subprocessos

```
countSubprocesses(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_SubProcess))->
    collect(oclAsType(FlowObjects_SubProcess))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os subprocessos contidos nesse caminho

```
seeSubprocesses(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
    Set(FlowObjects_SubProcess) =
        compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_SubProcess))->
        collect(oclAsType(FlowObjects_SubProcess)) -> asSet()
```

Context: class Model_Process

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de gateways

```
countGateways(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gateway))->
    collect(oclAsType(FlowObjects_Gateway))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as gateways contidas nesse caminho

```
seeGateways(origin:FlowObjects_FlowObject, destination:FlowObjects_FlowObject): Set(FlowObjects_Gateway)=
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gateway))->
    collect(oclAsType(FlowObjects_Gateway)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de gates

```
countGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate))->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as gates contidas nesse caminho

```
seeGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Set(FlowObjects_Gate) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de gates de input

```
countInputGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate))->select(type=#Input)->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as gates de input contidas nesse caminho

```
seeInputGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Set(FlowObjects_Gate) =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate)) ->select(type=#Input) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de gates de output

```
countOutputGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate))->select(type=#Output)->size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra as gates de output contidas nesse caminho

```
seeOutputGates(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject):Set(FlowObjects_Gate)=
    compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Gate))->
    collect(oclAsType(FlowObjects_Gate)) ->select(type=#Output) -> asSet()
```

Context: class Model_Process

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de eventos

```
countEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Event)) ->
  collect(oclAsType(FlowObjects_Event)) -> size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os eventos contidos nesse caminho

```
seeEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Set(FlowObjects_Event) =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_Event)) ->
  collect(oclAsType(FlowObjects_Event)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de start events

```
countStartEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_StartEvent)) ->
  collect(oclAsType(FlowObjects_StartEvent)) -> size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os start events contidos nesse caminho

```
seeStartEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
  Set(FlowObjects_StartEvent) =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_StartEvent)) ->
  collect(oclAsType(FlowObjects_StartEvent)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de intermediate events

```
countIntermediateEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_IntermediateEvent)) ->
  collect(oclAsType(FlowObjects_IntermediateEvent)) -> size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os intermediate events contidos nesse caminho

```
seeIntermediateEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
  Set(FlowObjects_IntermediateEvent) =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_IntermediateEvent)) ->
  collect(oclAsType(FlowObjects_IntermediateEvent)) -> asSet()
```

-- Função que percorre um dado caminho com uma origem e um destino e conta o número de end events

```
countEndEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) : Integer =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_EndEvent)) ->
  collect(oclAsType(FlowObjects_EndEvent)) -> size
```

-- Função que percorre um dado caminho com uma origem e um destino e mostra os end events contidos nesse caminho

```
seeEndEvents(origin: FlowObjects_FlowObject, destination: FlowObjects_FlowObject) :
  Set(FlowObjects_EndEvent) =
  compute(origin, destination) -> flatten -> select(oclIsKindOf(FlowObjects_EndEvent)) ->
  collect(oclAsType(FlowObjects_EndEvent)) -> asSet()
```

- **Funções Auxiliares**

Context: class Object

-- Função que calcula a potência

```
power(x: Real, n: Integer): Real =
  if (n=1) then
    x
  else
    x * power(x, n-1)
  endif
```

-- Função que calcula o logaritmo

```
logarithm(base: Real, x: Real): Real = ln(x) / ln(base)
```

```
ln(z: Real): Real = 2 * ln_aux(z, 0, 0, 0.0001)
```

```
pre: z >= 0
```

```
ln_aux(z: Real, n: Integer, previous: Real, delta: Real): Real =
```

```
  if (ln_sum(n+1, z) < delta) then
    previous
  else
    ln_aux(z, n+1, previous + ln_sum(n, z), delta)
  endif
```

```
ln_sum(n: Integer, z: Real): Real = 1 / (2*n+1) * power((z-1)/(z+1), 2*n+1)
```

Context: class FlowObjects_FlowObject

-- Função que retorna o connector que sai do FlowObject

```
flowSource(): Set(Connectors_SequenceFlow) = self.flowSource ->
  select(oclIsKindOf(Connectors_SequenceFlow)).oclAsType(Connectors_SequenceFlow) ->asSet()
```

-- Função que retorna o connector que entra no FlowObject

```
flowTarget(): Set(Connectors_SequenceFlow) = self.flowTarget ->
  select(oclIsKindOf(Connectors_SequenceFlow)).oclAsType(Connectors_SequenceFlow) ->asSet()
```

-- número total de conectores source de um FlowObject

```
totalNumberFlowSource(): Integer = flowSource()->size()
```

-- número total de conectores target de um FlowObject

```
totalNumberFlowTarget(): Integer = flowTarget()->size()
```

-- número total de conectores de um FlowObject

```
totalFlowObjectNumberArcs(): Integer = flowSource()->size() + flowTarget()->size()
```

-- Dado um flowObject, retorna os seus sucessores

```
successors(): Set(FlowObjects_FlowObject) = self.flowSource.target-> asSet()
```

Context: class FlowObjects_Gateway

-- Número de Joins - conta o número de Gates de Input de uma Gateway

numberInputGates() : Integer = self.gate-> select(type=#Input)->size()

-- Número de Splits - conta o número de Gates de Output de uma Gateway

numberOutputGates() : Integer = self.gate-> select(type=#Output)->size()

-- Verifica se a gateway é Join

isJoin(): Boolean = numberInputGates() > 1

-- Verifica se a gateway é Split

isSplit() : Boolean = numberOutputGates() > 1

-- Dada uma gateway, retorna as suas gates de saída

successors(): Set(FlowObjects_FlowObject) = self.gate -> select(type=#Output)

-- Dada uma gateway, retorna as suas gates de input

Inputsuccessors(): Set(FlowObjects_FlowObject) = self.gate -> select(type=#Input)

-- Dada uma gateway, retorna as suas gates

Allsuccessors(): Set(FlowObjects_FlowObject) = self.gate

Context: class FlowObjects_Gate

-- Dada uma gate, retorna o seu sucessor

```
successors(): Set(FlowObjects_FlowObject) =
    if type=#Input then
        oclEmpty(Set(FlowObjects_FlowObject))->including(self.gateway)
    else
        self.flowSource.target->asSet
    endif
```

Context: class StructuralElements_Pool

-- Conta as Lanes

numberLanes(): Integer = self.lane_swimlane -> select(oclIsKindOf(StructuralElements_Lane))-> collect(oclAsType(StructuralElements_Lane))->size()

-- Mostra as Lanes

showLanes(): Set(StructuralElements_Lane) = self.lane_swimlane -> select(oclIsKindOf(StructuralElements_Lane))-> collect(oclAsType(StructuralElements_Lane)) -> asSet()

Contex: class StructuralElements_Swimlane

-- Funções auxiliares para calcular todos os bpmnElements() numa swimlane

gateways(): Set(FlowObjects_Gateway) = self.flowObject->select(oclIsKindOf(FlowObjects_Gateway))->
collect(oclAsType(FlowObjects_Gateway))->asSet()

outGates(): Set(FlowObjects_Gate) = gateways().gate->select(type=#Output) -> asSet()

inGates(): Set(FlowObjects_Gate) = gateways().gate->select(type=#Input) -> asSet()

bpmnElementsAux(): Set(Model_BpmnElement) = self.flowObject ->collect(f: Model_BpmnElement | f) ->
union(
 self.flowObject.flowSource -> collect(f: Model_BpmnElement | f)) ->
union(
 self.outGates().flowSource -> collect(f: Model_BpmnElement | f)) -> asSet()

artifacts(): Set(Artifacts_Artifact) = self.bpmnElementsAux().associationSource.target ->
union (self.bpmnElementsAux().associationTarget.source) ->
select(oclIsKindOf(Artifacts_Artifact)) ->
collect(oclAsType(Artifacts_Artifact)) -> asSet()

activities(): Set(FlowObjects_Activity) = self.flowObject->select(oclIsKindOf(FlowObjects_Activity))->
collect(oclAsType(FlowObjects_Activity))->asSet()

events(): Set(FlowObjects_Event) = self.flowObject->select(oclIsKindOf(FlowObjects_Event))->
collect(oclAsType(FlowObjects_Event))->asSet()

messageFlows(): Set (Connectors_MessageFlow) = self.activities().messageFlowSource ->
union(
 self.events().messageFlowSource) ->
union(
 if (self.oclIsTypeOf(StructuralElements_Pool)) then
 self.oclAsType(StructuralElements_Pool).messageFlowSource
 else
 oclEmpty(Set (Connectors_MessageFlow))
 endif) -> asSet()

--Função que calcula todos os Elementos numa SwimLane

bpmnElements(): Set(Model_BpmnElement) = bpmnElementsAux() ->
union(
 self.artifacts() -> collect(f: Model_BpmnElement | f)) -> asSet()->
union(
 self.messageFlows() -> collect(f: Model_BpmnElement | f)) -> asSet()->
union (
 self.lane_swimlane ->
 iterate(elem: StructuralElements_Lane; acc: Set(Model_BpmnElement) =
 oclEmpty(Set(Model_BpmnElement)))|
 acc -> union(elem.bpmnElements()))

Bibliografia

- [Abreu, et al., 1994] Abreu, and Carapuça, "*Object-Oriented Software Engineering: Measuring and Controlling the Development Process*" in In Proceedings of the 4th International Conference on Software Quality, 1994.
- [Assaf Neto, 2003] Assaf Neto, Alexandre. "Finanças corporativas e valor". São Paulo: Atlas, 2003
- [Baroni, et al., 2002] Baroni, and Abreu: "*Formalizing Object-Oriented Design Metrics upon the UML Meta-Model*". In: Proc. of Brazilian Symposium on Software Engineering, Gramado - RS, Brazil (2002)
- [Baroni, et al., 2003] Baroni, and Abreu, "*A Formal Library for Aiding Metrics Extraction*" in International Workshop on Object-Oriented Re-Engineering at ECOOP'2003, Darmstadt, Germany, 2003.
- [Baroni, et al., 2004] Baroni, Calero, Ruiz, and Abreu: "*Formalizing Object-Relational Structural Metrics*". In: Proc. of 5ª Conferência da APSI, Lisbon (2004)
- [Bauerdick, Gogolla and Gutsche, 2004] Hanna Bauerdick, Martin Gogolla, and Fabian Gutsche. "*Detecting OCL Traps in the UML 2.0 Superstructure: An Experience Report*". In Baar et al. [BSMM04], pages 188–196.
- [Bento, 2009] <http://www3.uma.pt/bento/ppt/EstudosCorrelacionais.pdf>. Último acesso em 25 de Janeiro de 2009
- [Boehm, 1978] Boehm, B.W., et al., "*Characteristics of Software Quality*", North-Holland, 1978.
- [Boes, 2007] Boes, Klaas, Paul Leenards (2007). "Meer doen met minder mensen – Het scheiden van werkzaamheden in operations- en supportteams". *IT Service Management, best practices deel 2*. Zaltbommel (NL): Van Haren Publishing.
- [Bon and Hoving, 2007] J. van Bon, W. Hoving. "*Strategic Alignment Model Enhanced*". BHVB white paper, 2007.
- [Bon, 2008] Bon, J. v. (2008). "This is NOT IT Governance." *UPGRADE IX*(1): 9.
- [Botto, 2004] Botto, Renato (2004) "Arquitetura Corporativa de TI"; 2004
- [Brancheau, 1987] Brancheau, J. C., & Wetherbe, J. C. (1987). "Key Issues in Information Systems Management". *MIS Quarterly*.
- [Brenner, 2006] Brenner, M. (2006). "Classifying ITIL Processes - A Taxonomy under Tool Support Aspects." *IEEE* 10.
- [Brito e Abreu, et al., 1997] Brito e Abreu, Ochoa, and Goulão, "*The GOODLY Design Language for MOOD Metrics Collection*". INESC, Software Engineering Group, Technical Report R16/97, March, 1997.
- [Brito e Abreu, 2001] Brito e Abreu, "*Using OCL to Formalize Object-Oriented Design Metrics Definitions*". Software Engineering Group, INESC, Technical Report ES007/2001, May, 2001.
- [Broadbent and Weill, 1997] M. Broadbent and P. Weill. "*Management by Maxim: How business and IT managers can create IT infrastructures*". *Sloan Management Review*, Vol. 38, No. 3, Spring 1997, pp. 77–92
- [Broadbent and Weill, 1998] M. Broadbent & P. Weill. "*Leading Governance, Business and IT Processes*". ITEP Findings, 1998
- [Brown and Magill, 1998] – Brown, C. V. and Magill, S. L. "Reconceptualizing the context-design issue for the information systems function". *Organ. Sci.* 9 (2) 176-194
- [Brown, 1997] Brown, Carol V. "*Examining the Emergence of Hybrid Governance Solutions: Evidence from a Single Case Site*". *Information Systems Research* (8:1), Março 1997, pp. 69–94.
- [Brynjolfsson, 2000] Brynjolfsson, E., L. Hitt, et al. (2002). "*Intangible assets: How the interaction of computers and organizational structures affects stocks market valuations*". *Brookings Papers on Economic Activity: Macroeconomics* 1: 137–199
- [Caldeira, 2001] M. M. Caldeira, A. Serrano (2001). "Gestão de Investimentos em Sistemas e Tecnologias de Informação - uma revisão crítica". *Sistemas de Informação*, nº15. 99-107
- [Cardoso, 2006] Jorge Cardoso, J. Mendling, G. Neumann, H.A. Reijers, (2006). "A Discourse on Complexity of Process Models", BPI'06 - Second International Workshop on Business

- Process Intelligence, In conjunction with BPM 2006, Vienna, Austria, 5-7 September, 2006. J. Eder, S. Dustdar et al. (Eds.): BPM 2006 Workshops, LNCS 4103, pp. 115–126, 2006. Springer-Verlag, Berlin, Heidelberg, 2006.
- [Charan, Ram - 2005] Charan, Ram (2005) “From Compliance to Competitive Advantage”. ISBN-10: 0787971391
- [Cranefield, et al., 1999] Cranefield, and Purvis: “*UML as an Ontology Modelling Language*”. In: Proc. of Workshop on Intelligent Information Integration, 16th International Joint Conference on Artificial Intelligence (IJCAI-99) (1999)
- [Halstead, 1987] M. H. Halstead. “*Elements of software Science*”. Elsevier, Amsterdam, 1987.
- [Handy, 1992] Charles Handy. “*Balancing Corporate Power: A New Federalist Paper*”. Harvard Business Review, November–December 1992, pp. 59–72.
- [Capability Maturity Model, 2008] http://pt.wikipedia.org/wiki/Capability_Maturity_Model, último acesso 2009
- [Carnegie Mellon University, 2009] <http://www.sei.cmu.edu/cmmi/general/index.html>, último acesso 2009.
- [Costa, 2010] José Costa. “*MGPSI – Metodologia de Gestão de Projectos aplicada ao desenvolvimento de Sistemas de Informação*”, Dissertação de Mestrado sob orientação do Professor Doutor Fernando Brito e Abreu. 2010
- [Cyert, 1963] Cyert, R. M. and J. G. March (1963). “*A Behavioral Theory of the Firm*”. Englewood Cliffs, NJ: Prentice Hall.
- [Davenport, 1997] T.Davenport and L. Prusak - Information Politics.” *Information Ecology: Mastering the information and knowledge environment*. Oxford University Press, 1997.
- [Davenport and Metsisto, 1989] Thomas H. Davenport, Michael Hammer and Tauno J. Metsisto, “*How Executives Can Shape their Company's Information Systems*.” Harvard Business Review, March–April 1989, 130–134.
- [Davies, 1974] Davies, "An Experimental Investigation of Resource Allocation in Multiactivity Projects," 1974.
- [Debreceeny, 2006] Debreceeny, R. S. (2006). "Re-engineering IT Internal Controls: Applying Capability Maturity Models to the Evaluation of IT Controls." *IEEE*: 10.
- [Devaraj and Kohli, 2002] Sarv Devaraj and Rajiv Kohli. “*The IT Payoff: Measuring the Business Value of IT Investments*”. FT Prentice Hall, New Jersey USA, 2002
- [Dickson, 1984] Dickson, G. W., Leitheiser, R. L., Wetherbe, M., & Wetherbe, J. C. (1984). “Key Information System Issues for the 1980s”. *MIS Quarterly*.
- [Earl, 1993] M.J. Earl. “*Experiences in Strategic Information Systems Planning*”. *MIS Quarterly*, 17 (1993), pp. 1–24
- [Elmaghraby, et al., 1980] Elmaghraby, and Herroelen, "On the measurement of complexity in activity networks," *European Journal of Operational Research*, 1980.
- [Feigenbaum, 1951] Feigenbaum, Armand (1951). “*Quality Control: Principles, Practice, and Administration*” Chapter: "Total Quality Control".
- [FRC, 2005] <http://www.frc.org.uk/corporate/internalcontrol.cfm>, 2005, The Financial Reporting Council
- [Freitas, 2010] Jorge Freitas. “*Formalização de Níveis de Serviço na Gestão de Serviços de Tecnologias de Informação*”, Dissertação de Mestrado sob orientação do Professor Doutor Fernando Brito e Abreu. 2010
- [Godfrey, 1963] John Godfrey (1963), “*The Blind Men and the Elephant*”. Whittlesey House, New York.
- [Gogolla, Bohling and Richters, 2003] Martin Gogolla, Jörn Bohling, and Mark Richters. “*Validation of UML and OCL Models by Automatic Snapshot Generation*”. In Grady Booch, Perdita Stevens, and Jonathan Whittle, editors, *Proceedings of the 6th International Conference on Unified Modeling Language (UML'2003)*, pages 265–279, San Francisco, CA, USA, 2003. Springer, Berlin, LNCS 2863.
- [Gogolla, Bohling and Richters, 2005] Martin Gogolla, Jörn Bohling, and Mark Richters. “*Validating UML and OCL Models in USE by Automatic Snapshot Generation*”. *Journal on Software and System Modeling*, 4(4):386–398, 2005.
- [Ghani, 2008] A. Ghani, K. Muketha, W. Wen. “*Complexity Metrics for Measuring the Understandability and Maintainability of Business Process Models using Goal-Question-Metric (GQM)*”. *International Journal of Computer Science and Network Security*, VOL.8 No.5, Maio 2008

- [Goulão, et al., 2004a] Goulão, and Abreu: “*Formalizing Metrics for COTS*”. In: Proc. of International Workshop on Models and Processess for the Evaluation of COTS Components (MPEC 2004) at ICSE 2004, Edimburgh, Scotland, pp. 37-40. IEE (2004a)
- [Goulão, et al., 2004b] Goulão, and Abreu: “*Cross-Validation of a Component Metrics Suite*”. In: Proc. of IX Jornadas de Ingeniería del Software y Bases de Datos, Málaga, Spain (2004b)
- [Goulão, et al., 2005a] Goulão, and Abreu: “*Formal Definition of Metrics upon the CORBA Component Model*”. In: Proc. of First International Conference on the Quality of Software Architectures, QoSA'2005, Erfurt, Germany. Springer (2005a)
- [Goulão, et al., 2005b] Goulão, and Abreu: “*Composition Assessment Metrics for CBSE*”. In: Proc. of 31st Euromicro Conference - Component-Based Software Engineering Track, Porto, Portugal. IEEE Computer Society (2005b)
- [Henry and Kafura, 1981] S. Henry, D. Kafura, (1981). “*Software Structure Metrics based on Information-flow*” IEEE Transactions on Software Engineering, 7(5), pp.510 – 518
- [Hoffman and Weill, 2007] F. G.-M. H. P. (2007). “*Banknorth: Designing IT Governance for a Growth-Oriented Business Environment*.” CISR WP No. 350: 20.
- [IDEF, 2006] <http://www.idef.com/>, 2006
- [International Organization for Standardization, 2005] International Organization for Standardization, ISO/IEC 20000-1 & ISO/IEC 20000-2, 2005.
- [Iludícibus, 1998] – Iludícibus, Sérgio. “*Análise de Balanços*”. Ed. São Paulo: Atlas, 1998. 7.
- [International Standard, 2008] International Standard ISO/IEC 38500 - Corporate Governance of information technology. 2008.
- [ISACA, 2005] http://www.isaca.org/Content/NavigationMenu/Members_and_Leaders1/COBIT6/COBIT_Publications/COBIT_Products.htm, 2009
- [ISO/IEC Standards, 2008] http://www.iso.org/iso/iso_catalogue.htm
- [ISO 15504, 2009] http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_tc_browse.htm?commid=45020&published=on&includesc=true
- [ISO 15504, 2008] <http://www.isospice.com/categories/ISO%7B47%7DIEC-15504-Standard/>
- [ISO 15504, 2008] http://pt.wikipedia.org/wiki/ISO/IEC_15504, 2008
- [IT Governance Institute, 2000] IT Governance Institute “COBIT 3rd Edition” Síntese de Julho de 2000
- [ITIL v3, 2007] Iqbal, Majid and Nieves, Michael (2007). “Service Strategy”. TSO (The Stationery Office), United Kingdom.
- [Johnson, 2008] Johnson, M. S. a. P. (2008). “The IT organization modeling and assessment tool: Correlating IT governance maturity with the effect of IT.” 10.
- [Kaimann, 1974] Kaimann, “*Coefficient of network complexity*”. Management Science, vol. 21, pp. 172-177, 1974.
- [Kaimann, 1975] Kaimann, “*Coefficient of network complexity: Erratum*”. Management Science, vol. 21, pp. 1211-1212 1975.
- [keen, 1991] Peter G.W. Keen. “*Shaping the Future: Business Design Through Information Technology*”. Harvard Business School Press, Cambridge, MA, 1991
- [keen, 1995] Peter G.W. Keen. “*Every Manager’s Guide to Information Technology*”. 2nd Edition, Boston, MA: Harvard Business School Press, 1995.
- [McCabe, 1976] McCabe, “*A complexity measure*” IEEE Transactions on Software Engineering, vol. 4, pp. 308-320, December 1976.
- [Lepasar and Makinen, 2002] Lepasar and Makinen, M. (2002). “Integrating Software Process Assessment Models using a Process Meta Model.” IEEE: 6.
- [Malva, 2008] <http://www.estv.ipv.pt/PaginasPessoais/malva/MetodosElectro/Regress%C3%A3o.pdf>
- [Mansur, 2007] Ricardo Mansur. “Governança de TI: Metodologias, Frameworks e Melhores Práticas”. 224 Páginas. 1ª Edição – 2007. ISBN: 9788574523224
- [Marcus, 2000] Markus, M. L., Axline, S., Petrie, D., & Tanis, S. C. (2000). “Learning from adopters’ experiences with ERP: problems encountered and success achieved”. Journal of Information Technology, 245-265
- [McGuinness, et al., 2004] McGuinness, and Harmelen, “*OWL Web Ontology Language Overview*”. W3C, Recommendation 10 February, 2004.

- [McKeen and Smith, 2003] James D. McKeen and Heather A. Smith. *"Making IT Happen: Critical Issues in IT Management"*. Chichester, West Sussex, England: Wiley, 2003.
- [McLean et al., 1977] McLean, E.R. e J.V. Soden, *"Strategic Planning for MIS"*, John Wiley, New York, 1977.
- [Mendling, 2007] Jan Mendling, (2007). *"Testing Density as a Complexity Metric for EPCs"*. Vienna University of Economics and Business Administration Augasse 2-6, A-1090Wien, Austria.
- [Nicho and Cusack, 2007] Cusack, M. Nicho, B. (2007). "A Metrics Generation Model for Measuring the Control Objectives of Information Systems Audit." IEEE - Computer Society: 11.
- [OCL, 2003] OMG UML 2.0 OCL specification, Oct 2003. Disponível em <http://www.omg.org/cgi-bin/apps/doc?ptc/03-10-14.pdf>
- [O'Connor, 1993] O'Connor, D. Anne (1993), "Successful strategic information systems planning". *Journal of Information Systems*, 3, 71-83.
- [OECD, 1999] OECD – Organization for Economic Co-operation and Development. (1999). "OECD Principles of Corporate Governance".
- [Office of Government Commerce, 2003] Office of Government Commerce, IT Infrastructure Library Service Delivery, The Stationery Office, 2003.
- [Oliveira, 1994] Oliveira, A. (1994) *"O Valor da Informação"*. In Dossier Informação. Revista Pequena e Média Empresa, nº12-3ª Série.
- [Oxford University Press, 2003] Oxford English Dictionary, Oxford University Press (OUP), Second Edition 1989, http://dictionary.oed.com/cgi/entry/00070944?single=1&query_type=word&queryword=duopoly&edition=2e&first=1&max_to_show=10 (accessed 30 May 2003).
- [Pascoe, 1966] Pascoe, "Allocation of resources-CPM," *Revue Française de Recherche Opérationnelle*, vol. 38, pp. 31-38, 1966.
- [Paulk et al, 1995] Mark C. Paulk, M. D. K., and Suzanne M. Garcia (1995). "CMM Versus SPICE Architectures." *IEEE Software Process Newsletter* No. 3: 6.
- [Peppard, 2002] Ward, J., Peppard, J., (2002). *"Strategic Planning for Information Systems"*, 3rd ed., John Wiley & Sons.
- [Pereira, 2007] Pereira, João Carlos (2007). "Aplicabilidade de um Framework para a Governança de TI".
- [Peterson, 2004] Peterson, Ryan(2004)"Crafting Information Technology Governance". *EDPACS*, 32:6, 1 — 24
- [Peterson, 2001] Peterson, R. R. (2001). *"Information governance: an empirical investigation into the differentiation and integration of strategic decision-making for IT"*. Unpublished paper, Tilburg University, The Netherlands.
- [Piessens, Frank, 2002] "On the importance of the separation-of-concerns principle in secure software engineering". 10.
- [Pitt, 2001] <http://www.sec.gov/news/speech/spch530.htm>, 2001, Chairman Harvey L. Pitt
- [Popovic, 2007] Spremic, M., Popovic, M., 2007. "Towards a Corporate IT Risk Management Model, *Proceedings of the 6th WSEAS International Conference on Information Security and Privacy*". Puerto de la Cruz, Tenerife, Canary Islands, Spain, December 2007, pp. 111-7.
- [Richters and Gogolla, 2003] Mark Richters and Martin Gogolla. *"Aspect-oriented monitoring of UML and OCL constraints"*. In Omar Aldawud, Mohamed Kande, Grady Booch, Bill Harrison, Dominik Stein, Jeff Gray, Siobhan Clarke, Aida Zakaria, Peri Tarr, and Faisal Akkawi, editors, *Proceedings of the UML 2003 Workshop Aspect-Oriented Software Development with UML*. Illinois Institute of Technology, Department of Computer Science, <http://www.cs.iit.edu/oaldawud/AOM/index.htm>, 2003.
- [Rockart, 1996] Rockart, J.F., M.J. Earl, and J.W. Ross (1996) *"Eight Imperatives for the New IT Organization"*, *Sloan Management Review* (38)1, pp. 43-56.
- [Ross, 2003] Jeanne Ross *"Creating a Strategic IT Architecture Competency: Learning in Stages"*. MISQE, Vol. 2, No.1, March 2003.
- [Ross and Beath, 2002] Jeanne W. Ross and Cynthia M. Beath. *"Beyond the Business Case: New Approaches to IT Investment"*. MIT Sloan Management Review, Vol. 43 No. 2 (Winter 2002), pp. 51–59.
- [Rozemeijer, 2007] Rozemeijer, E., Bon, J.V., Verheijen, T. (2007). *"Framework for IT Management - A Pocket Guide"*. itSMF International. Van Haren Publishing.
- [Sallé, 2004] M.Sallé, (2004). "IT Service Management and IT Governance: Review, Comparative Analysis and their Impact on Utility Computing", HP Labs Technical Report HPL-2004-98.

- [Sambamurthy and Zmud, 1999] Sambamurthy, V. and R. W. Zmud (1999). "Arrangements for Information Technology Governance: A Theory of Multiple Contingencies", *MIS Quarterly*, 23(2): 261–290.
- [Sarbanes Oxley, 2002] <http://en.wikipedia.org/wiki/Sarbanes-oxley>, 2002
- [Simonsson and Johnson, 2008] Simonsson, Marten and Johnson, Pontus - "The IT organization modeling and assessment tool: Correlating IT governance maturity with the effect of IT". IEEE.10.
- [Simon and Barnard, 1961] Simon, H. A. and C. I. Barnard (1961). "Administrative Behavior: A Study of Decision-Making: Processes in Administrative Organization". New York: Macmillan.
- [Smith, 2003] Smith, "Ontology," in Blackwell Guide to the Philosophy of Computing and Information, Floridi, L., Ed. Oxford: Blackwell Scientific Publications, 2003, pp. 155-166.
- [SINFIC, 2005] "Principais Tendências em Modelos de Qualidade". Newsletter nº30
- [Sohal and Fitzpatrick] A.S. Sohal, P. Fitzpatrick. "IT governance and management in large Australian organizations". *International Journal of Production Economics*, 75, 94-112, 2002.
- [Solingen, 1999] Solingen, R., Berghout, E (1999). "The Goal/Question/Metric Method: A Practical Guide for Quality Improvement of Software Development". McGraw-Hill.
- [Spremic et al, 2008] Mario Spremic, Z. Zmirak, Krunoslav Kraljevic (2008). "IT and Business Process Performance Management: Case Study of ITIL Implementation in Finance Service Industry." (Proceedings of the ITI 2008 30th Int. Conf. on Information Technology Interfaces, June 23-26, 2008, Cavtat, Croatia): 8.
- [Sraeel, 2004] Sraeel, H. (2004). "Taking a Closer Look into IT Governance Globally". *Bank Technology News*, 8
- [The University of British Columbia, 2009] <http://www.e-strategy.ubc.ca/news/update0309/bpr.html>, 2009, The University of British Columbia.
- [Standards Australia, 2005] <http://www.standards.org.au/>
- [UML, 2005] OMG UML 2.0 superstructure specification, August 2005. Disponível em <http://www.omg.org/cgi-bin/apps/doc?formal/05-07-04.pdf>.
- [Vanhoucke, et al., 2008] Vanhoucke, Coelho, Debels, Maenhout, and Lv, "An evaluation of the adequacy of project network generators with systematically sampled networks". *European Journal of Operational Research*, 2008.
- [Volker, 2007] Volker Gruhn, Ralf Laue, (2007). "2 Approaches for Business Process Model Complexity Metrics". *Technology for Business Information System*, pp.13- 24 @ 2007 Springer.
- [Weill, 2006] Weill, Peter e Ross, Jeanne W (2006) "IT Governance".
- [Weill and Ross, 2004] Weill, Peter e Ross, Jeanne W (2004) "IT Governance on One Page". CISR WP Nº 349.
- [Weill, 2004] "Don't just lead, Govern: How Top-Performing Firms Govern IT." CISR WP Nº 341: 21.
- [Weill and Broadbent, 2003] Marianne Broadbent and Peter Weill, "Effective IT Governance. By Design" January 2003. Gartner EXP Premier Report
- [Weill and Woodham, 2002] Peter Weill and Richard Woodham. "Don't Just Lead, Govern: Implementing Effective IT Governance." CISR Working Paper No. 326: 17
- [Weill, Subramani and Broadbent, 2002] Peter Weill, Mani Subramani and Marianne Broadbent "Building IT Infrastructure for Strategic Agility". *Sloan Management Review*, Vol. 44, No 1 (Fall 2002), pp. 57–65.
- [Weill and Gartner, 2002] Peter Weill and Gartner. Inc. – "Creating Effective IT Governance" - U.S. Symposium/ITxpo - 2002
- [Weill and Broadbent, 1998] Peter Weill and Marianne Broadbent, "Leveraging the New Infrastructure: How market leaders capitalize on IT" (Boston, MA: Harvard Business School Press, 1998), Chapter 3.
- [Wendy, et al., 1994] Wendy, and Dolores, *Software Error Analysis*: Silicon Press, 1994.
- [Wendy, et al., 1995] Wendy, and Dolores, *Software Error Analysis*: Silicon Press, 1995.
- [wikipédia, 2009] http://en.wikipedia.org/wiki/Business_Process_Management, 2009
- [wikipédia, 2009] http://en.wikipedia.org/wiki/Design_for_Six_Sigma, 2009
- [wikipédia, 2009] http://en.wikipedia.org/wiki/Six_Sigma, 2009
- [Wikipédia, Economies of scale] http://en.wikipedia.org/wiki/Economies_of_scale, último acesso em 10 de Junho de 2009
- [Wikipédia, Enron_scandal] http://en.wikipedia.org/wiki/Enron_scandal, 2009

- [Wikipédia, Estatística] <http://pt.wikipedia.org/wiki/Estat%C3%ADstica>, último acesso em 13 de Janeiro de 2010
- [Wikipédia, GQM] <http://en.wikipedia.org/wiki/GQM>, último acesso em 06 de Setembro de 2009
- [Wikipédia, Moodle] <http://pt.wikipedia.org/wiki/Moodle>, último acesso em 11 de Junho de 2009
- [Wikipédia, Pearson] http://pt.wikipedia.org/wiki/Karl_Pearson, último acesso em 26 de Janeiro de 2010
- [Wikipédia, Trade-off] <http://pt.wikipedia.org/wiki/Trade-off>, último acesso em 9 de Junho de 2009
- [wikipédia, Visual Basic for Applications] http://en.wikipedia.org/wiki/Visual_Basic_for_Applications, último acesso em 23 de Novembro de 2009
- [Wim Van Grembergen, 2007] Wim Van Grembergen "*Introduction to the minitrack: IT Governance and its mechanisms*". 35th HICSS conference
<http://computer.org/proceedings/hicss/1874/track8/187480242.pdf>).
- [Woodhouse, 2007] S. Woodhouse, 2007. "Information Security: End User Behavior and Corporate Culture," in *CIT2007 IEEE 7th International Conference on Computer and Information Technology* University of Aizu, Fukushima Japan.
- [Woodhouse, 2008] Woodhouse, S. (2008). "An ISMS (Im)-Maturity Capability Model." IEEE 8th International Conference on Computer and Information Technology Workshops: 6.
- [Zmud, 2000] Robert W. Zmud, ed. "*Framing the Domains of IT Management: Projecting the Future...Through the Past*". Cincinnati: Pinnaflex, 2000.
- [Zviran, 2003] Zviran, M. (2003). "User Satisfaction in ERP Systems: Some Empirical Evidence". Journal of the Academy of Business and Economics, March, 3.